

Desagregación de Estimaciones en Áreas
Pequeñas un enfoque bayesiano
CEPAL - Unidad de Estadísticas Sociales

Table of contents I

Objetivo de desarrollo sostenible

Limitaciones de las encuestas.

Uso de métodos SAE

Función Generalizada de Varianza (FGV)

Modelos de área.

Modelos de unidad.

Modelo de unidad Índice de Privación Multidimensional (IPM)

Table of contents II

Modelo de área para estadísticas del mercado de trabajo

Objetivo de desarrollo sostenible



OBJETIVOS DE DESARROLLO SOSTENIBLE

1 FIN
DE LA POBREZA



2 HAMBRE
CERO



3 SALUD
Y BIENESTAR



4 EDUCACIÓN
DE CALIDAD



5 IGUALDAD
DE GÉNERO



6 AGUA LIMPIA
Y SANEAMIENTO



7 ENERGÍA ASEQUIBLE
Y NO CONTAMINANTE



8 TRABAJO DECENTE
Y CRECIMIENTO
ECONÓMICO



9 INDUSTRIA,
INNOVACIÓN E
INFRAESTRUCTURA



10 REDUCCIÓN DE LAS
DESIGUALDADES



11 CIUDADES Y
COMUNIDADES
SOSTENIBLES



12 PRODUCCIÓN
Y CONSUMO
RESPONSABLES



13 ACCIÓN
POR EL CLIMA



14 VIDA
SUBMARINA



15 VIDA
DE ECOSISTEMAS
TERRESTRES



16 PAZ, JUSTICIA
E INSTITUCIONES
SÓLIDAS



17 ALIANZAS PARA
LOGRAR
LOS OBJETIVOS



OBJETIVOS
DE DESARROLLO
SOSTENIBLE

Algunas metas del ODS2 (Hambre cero)

De aquí a 2030, poner fin al hambre y asegurar el acceso de todas las personas, en particular los pobres y las personas en situaciones de vulnerabilidad, incluidos los niños menores de 1 año, a una alimentación sana, nutritiva y suficiente durante todo el año.

- ▶ Prevalencia de la subalimentación.
- ▶ Prevalencia de la inseguridad alimentaria moderada o grave en la población, según la Escala de Experiencia de Inseguridad Alimentaria.

Algunas metas del ODS8 (Empleo decente)

De aquí a 2030, lograr el empleo pleno y productivo y el trabajo decente para todas las mujeres y los hombres, incluidos los jóvenes y las personas con discapacidad, así como la igualdad de remuneración por trabajo de igual valor.

- ▶ Tasa de desempleo, desglosada por sexo, edad y personas con discapacidad.

Principio fundamental de la desagregación de datos

Los indicadores de los Objetivos de Desarrollo Sostenible deberán desglosarse, siempre que sea pertinente, por ingreso, sexo, edad, raza, etnicidad, estado migratorio, discapacidad y ubicación geográfica, u otras características, de conformidad con los Principios Fundamentales de las Estadísticas Oficiales.

Resolución de la Asamblea General - 68/261

Limitaciones de las encuestas.

¿Qué es el coeficiente de variación?

El coeficiente de variación es una medida de error relativo a un estimador, se define como:

$$cve(\hat{\theta}) = \frac{se(\hat{\theta})}{\hat{\theta}}$$

Muchas veces se expresa como un porcentaje, aunque no está acotado a la derecha, y por eso es conveniente a la hora de hablar de la precisión de una estadística que viene de una encuesta.

Estándares de alerta en algunos países (encuestas de hogares)

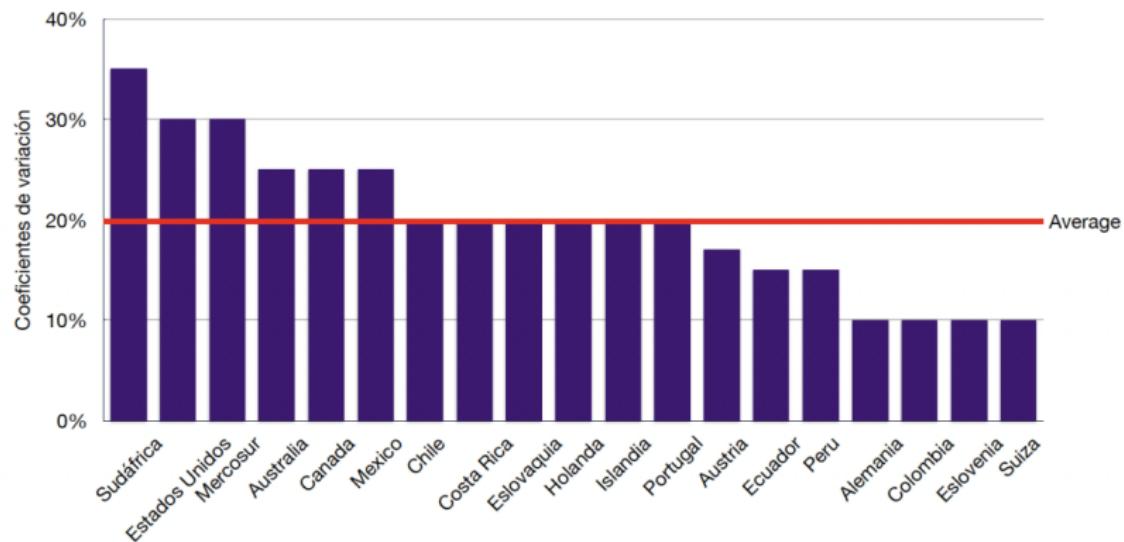


Figure 1: Alertas sobre los coeficientes de variación

Algunas alertas definidas en la publicación

Cuando se sobrepasa el umbral del coeficiente de variación aparecen algunas de las siguientes alertas:

- ▶ No se publica
- ▶ Usar con precaución.
- ▶ Las estimaciones requieren revisiones, no son precisas y se deben usar con precaución.
- ▶ Poco confiable, menos preciso.
- ▶ No cumple con los estándares de publicación.
- ▶ Con reserva, referencial, cuestionable.
- ▶ Valores muy aleatorios, estimación pobre.

Dominios de estudio y subpoblaciones de interés

Una encuesta se planea con el fin de generar información precisa y confiable en los dominios de estudio que se han predefinido. Sin embargo, existen subgrupos poblacionales que la encuesta no abordó en su diseño, y sobre los cuales se quisiera una mayor precisión.

- ▶ Incidencia de la pobreza desagregado por departamento o provincia (tamaño de muestra conocido y planificado).
- ▶ Tasa de desocupación desagregada por sexo (tamaño de muestra aleatorio, pero planificado).
- ▶ Tasa de asistencia neta estudiantil en primaria desagregada por quintiles de ingreso (tamaño de muestra aleatorio).

Precisión de los estimadores

Debido a que una encuesta es una investigación parcial sobre una población finita, es necesario saber que:

- ▶ A partir de una encuesta, no se calculan indicadores, sino que se estiman con ayuda de los datos de la encuesta.
- ▶ Es necesario calcular el grado de error que se comete al no poder realizar una investigación exhaustiva. Este error es conocido como el error de muestreo.
- ▶ La precisión de un estimador está supeditada al intervalo de confianza.

Entre más angosto sea el intervalo, más precisión se genera y por ende se tiene un menor error de muestreo.

El tamaño de muestra efectivo

- ▶ En las encuestas de hogares, con diseños de muestreo complejos, no existe una sucesión de variables que sean independientes e identicamente distribuidas.
- ▶ La muestra y_1, \dots, y_n no es un vector en el espacio n -dimensional, donde se asume que cada componente del vector puede variar por sí mismo.
- ▶ La dimensión final del vector (y_1, \dots, y_n) es mucho menor que n , puesto que existe una forma jerárquica en la selección de los hogares y a la interrelación de la variable de interés con las UPMs

El tamaño de muestra efectivo

El tamaño de muestra efectivo se define como sigue:

$$n_{efectivo} = \frac{n}{Deff}$$

En donde Deff es el efecto de diseño que depende de: 1. El número de encuestas promedio que se realizaron en cada UPM. 2. La correlación existente entre la variable de interés y las mismas UPMs.

Es posible considerar que, si el tamaño de muestra efectivo no es mayor a un umbral, entonces la cifra no debería ser considerada para publicación.

Grados de libertad

En las subpoblaciones los grados de libertad no se consideran fijos sino variables.

$$gl = \sum_{h=1}^H v_h \times (n_{Ih} - 1)$$

Note que v_h es una variable indicadora que toma el valor uno si el estrato h contiene uno o mas casos de la subpoblación de interés, n_{Ih} es el número de UPMs en el estrato. En el caso más general, los grados de libertad se reducen a la siguiente expresión:

$$gl = \#UPMs - \#Estrato$$

Uso de métodos SAE

Justificación

- ▶ Los estimadores directos, basados solo en unidades de muestreo observadas para cada área pequeña, no son suficientemente confiables.
- ▶ Tamaño de muestra pequeño o incluso ninguna unidad observada (falta de información).
- ▶ El coeficiente de variación (CV) es demasiado alto para el indicador objetivo a nivel de área.

Incremento del coeficiente de variación

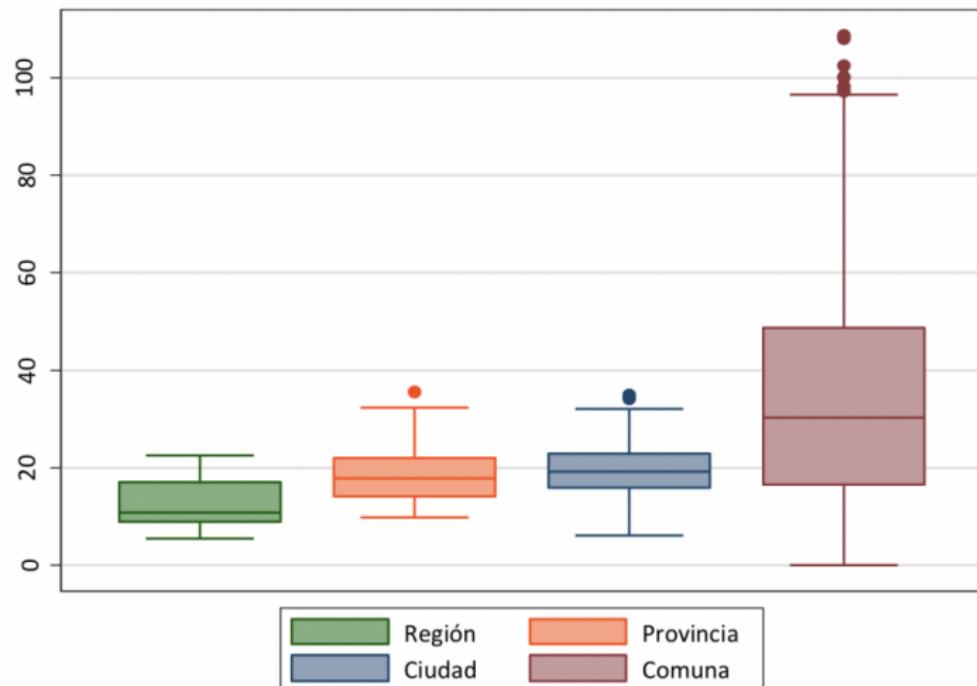


Figure 2: Distribución de los coeficientes de variación en Chile

Justificación

Cuando los estimadores directos no son confiables para algunos dominios de interés, existen dos opciones:

- 1 Sobremuestreo: aumentar el tamaño de la muestra en los dominios de interés (aumento de los costos).
- 2 Aplicar técnicas estadísticas que permitan estimaciones confiables en esos dominios, métodos SAE.

¿Qué es un área pequeña?

- ▶ La mayoría de las encuestas nacionales están planificadas para entregar estimaciones confiables a nivel nacional y regional pero a niveles más bajos se reduce la precisión.
- ▶ Un área pequeña es un dominio para el cual el tamaño de muestra específico no es suficientemente grande para obtener estimaciones confiables.
- ▶ Habitualmente son dominios no planificados y su tamaño de muestra esperado es aleatorio y es más grande a medida que aumenta el tamaño de la población del área.

¿Qué es un área pequeña?

La subpoblación de interés puede ser una zona geográfica o subgrupos socioeconómicos.

- ▶ Geográfico: provincias, áreas del mercado de trabajo, municipios, sectores censales para medir por ejemplo la tasa de desempleo a nivel comunal.
- ▶ Dominio de subgrupos específicos: edad × sexo × raza dentro del ámbito geográfico de una zona, para medir por ejemplo la tasa de desempleo por sexo o edad específica en las zonas urbanas.

Algunos métodos

- ▶ Los estimadores SAE se dividen en dos tipos principales dependiendo de cómo se aplican los modelos a los datos dentro de las áreas pequeñas: nivel de área y nivel de unidad.
- ▶ Los estimadores de área pequeña se basan en cálculos de nivel de área si los modelos vinculan la variable de interés y con variables auxiliares x específicas del área.

Algunos métodos

- ▶ Se llaman modelos a nivel de unidad si se vinculan valores individuales para las variables auxiliares específicas de la unidad.
- ▶ Los estimadores basados en áreas pequeñas se calculan a nivel de área si los datos de la unidad no están disponibles.
- ▶ También pueden ser calculados si los datos de nivel de unidad están disponibles resumiéndolos en el nivel de área apropiado.

Proceso de estimación



Figure 3: Producción de estadísticas con SAE

Consideraciones

- ▶ Todos los métodos SAE requieren datos auxiliares a nivel del área pequeña desde el cual toman prestada la fuerza.
- ▶ La efectividad de los métodos SAE depende del grado de asociación entre la variable de interés y los datos auxiliares.
- ▶ La búsqueda de buenas variables auxiliares es crítica, incluida la construcción imaginativa de tales variables.
- ▶ Los datos auxiliares deben medirse de manera consistente a través de las áreas pequeñas, pero pueden incluir estimaciones de muestras grandes con error de muestreo conocido.

Desafíos

- ▶ Aumento de las tasas de no respuesta.
- ▶ Aumento de costos, menos financiación.
- ▶ Aumento de la demanda de estimaciones para dominios pequeños como por raza, etnia o pobreza.
- ▶ Aumento de la demanda de estimaciones de áreas pequeñas.
- ▶ Aumento de la complejidad en los contenidos de los cuestionarios y por lo tanto la carga de respuesta.
- ▶ Aumento de la demanda de análisis secundarios, uso público y archivos de datos de uso restringido.

Función Generalizada de Varianza (FGV)

¿Cuál es la importancia de la Función Generalizada de Varianza?

- ▶ La varianza del estimador directo es un insumo crucial en el modelo de áreas.
- ▶ No es posible calcular la varianza del estimador directo a nivel de dominio.
- ▶ En dominios con un tamaño de muestra muy pequeño, las estimaciones de varianza pueden ser poco fiables.
- ▶ Se sugiere la utilidad de un modelo de suavizamiento de las varianzas.
- ▶ El propósito del suavizamiento es eliminar el ruido y la volatilidad en las estimaciones de varianza para obtener una señal más precisa del proceso.

La Función Generalizada de Varianza

Hidiroglou (2019) establece que: $E_{\mathcal{MP}}(\hat{\theta}_d^{dir}) = x_d^T \beta$ y $V_{\mathcal{MP}}(\hat{\theta}_d^{dir}) = \sigma_u^2 + \tilde{\sigma}_d^2$, en donde el subíndice \mathcal{MP} hace referencia a la inferencia doble que se debe tener en cuenta en este tipo de ajustes.

- ▶ \mathcal{M} hace referencia a la medida de probabilidad inducida por el modelamiento y la inclusión de las covariables auxiliares (x_d).
- ▶ \mathcal{P} hace referencia a la medida de probabilidad inducida por el diseño de muestreo complejo que induce las estimaciones directas.

Estimación de la Varianza de Muestreo

La FGV consiste en ajustar un modelo log-lineal a la varianza directa estimada. Partiendo del hecho de que se tiene acceso a un estimador insesgado de σ^2 , denotado por $\hat{\sigma}^2$ se tiene que:

$$E_{\mathcal{MP}}(\hat{\sigma}_d^2) = E_{\mathcal{M}}(E_{\mathcal{P}}(\hat{\sigma}_d^2)) = E_{\mathcal{M}}(\sigma_d^2) = \tilde{\sigma}_d^2$$

La anterior igualdad puede interpretarse como que un estimador insesgado y simple de $\tilde{\sigma}_d^2$ puede ser $\hat{\sigma}_d^2$.

Modelos de Suavizamiento

Rivest y Belmonte (2000) proponen modelos de suavizamiento para estimar las varianzas directas. Estos modelos se definen de la siguiente manera:

$$\log(\hat{\sigma}_d^2) = z_d^T \alpha + \varepsilon_d$$

En donde z_d es un vector de covariables explicativas que son funciones de x_d , α es un vector de parámetros que deben ser estimados, ε_d son errores aleatorios con media cero y varianza constante, que se asumen idénticamente distribuidos condicionalmente sobre z_d .

Estimación Suavizada

- La estimación suavizada de la varianza de muestreo está dada por:

$$\tilde{\sigma}_d^2 = E_{\mathcal{MP}} (\sigma_d^2) = \exp (z_d^T \alpha) \times \Delta$$

En donde, $E_{\mathcal{MP}} (\varepsilon_d) = \Delta$.

- Haciendo uso del método de los momentos, se tiene el siguiente estimador insesgado para Δ :

$$\hat{\Delta} = \frac{\sum_{d=1}^D \hat{\sigma}_d^2}{\sum_{d=1}^D \exp (z_d^T \alpha)}$$

Estimación de parámetros

- La estimación del coeficiente de parámetros de regresión está dada por la siguiente expresión:

$$\hat{\alpha} = \left(\sum_{d=1}^D z_d z_d^T \right)^{-1} \sum_{d=1}^D z_d \log(\hat{\sigma}_d^2)$$

- Y el estimador suavizado de la varianza muestral está definido por:

$$\hat{\tilde{\sigma}}_d^2 = \exp(z_d^T \hat{\alpha}) \hat{\Delta}$$

Datos: Gran Encuesta Integrada de Hogares (GEIH) de Colombia.

La Gran Encuesta Integrada de Hogares (GEIH) del 2018 en Colombia, utilizó un diseño muestral complejo que incluyó la estratificación de la población en zonas urbanas y rurales, junto con un muestreo por conglomerados. La muestra seleccionada fue significativa, permitiendo la recolección de datos de manera representativa en todo el país. En total, se utilizaron 98,000 Unidades Primarias de Muestreo (UPM) para tener estadísticas confiables a Nivel Nacional, Regiones Geográficas, Ciudades principales y Áreas Urbanas/Rurales, Estratos Socioeconómicos.

Set de datos

Table 1: GEIH Colombia

dam	dam2	wkx	upm	estrato	pobreza
05	05360	127.2	010126005360	051	0
05	05360	127.2	010126005360	051	0
05	05360	127.2	010126005360	051	0
05	05360	127.2	010126005360	051	0
05	05360	127.2	010126005360	051	0
05	05360	125.7	010126005360	051	1
05	05360	125.7	010126005360	051	1
05	05360	125.7	010126005360	051	1
05	05360	125.7	010126005360	051	1
05	05360	115.9	010126005360	051	1

Diseño muestral

Para definir el diseño muestral a partir de una base de datos de encuesta se usan las librerías `survey` y `srvyr`.

```
library(survey)
library(srvyr)
options(survey.lonely.psu = "adjust")

diseno <-
  as_survey_design(
    ids = upm,
    weights = wkx,
    strata = estrato,
    nest = TRUE,
    .data = encuesta
  )
```

Estimaciones directas por dominio

Para la estimación directa de la proporción se emplea la función `direct.supr`, disponible en el archivo `0Source_FH.R`. Esta función realiza las estimaciones y criterios de calidad en una encuesta de muestreo complejo con diseño estratificado y por conglomerados.

```
directodam2 <- direct.supr(design.base = diseno,
                             variable = pobreza,
                             group = dam2,
                             upm = upm,
                             estrato = estrato)
```

Dominios seleccionados

- ▶ Mínimo 50 observaciones por dominio.
- ▶ Efecto de diseño (Deff) mayor a 1.
- ▶ Mínimo 3 grados de libertad.

Table 2: Conteo de dominios seleccionados

Flag	n
Excluir	59
Incluir	379

FGV para la GEIH de Colombia

Para este proceso se realiza la transformación $\log(\hat{\sigma}_d^2)$ y la selección de las columnas identificador del municipio (dam2), la estimación directa (pobreza), el número de personas en el dominio (nd) y la varianza estimada (vardir).

Table 3: Set datos para la FGV

dam2	pobreza	nd	vardir	ln_sigma2
05001	0.1597	27432	0.0000	-10.012
05002	0.4049	257	0.0032	-5.737
05031	0.3817	199	0.0042	-5.463
05034	0.4731	223	0.0018	-6.335
05045	0.2876	480	0.0064	-5.045

Analisis gráfico

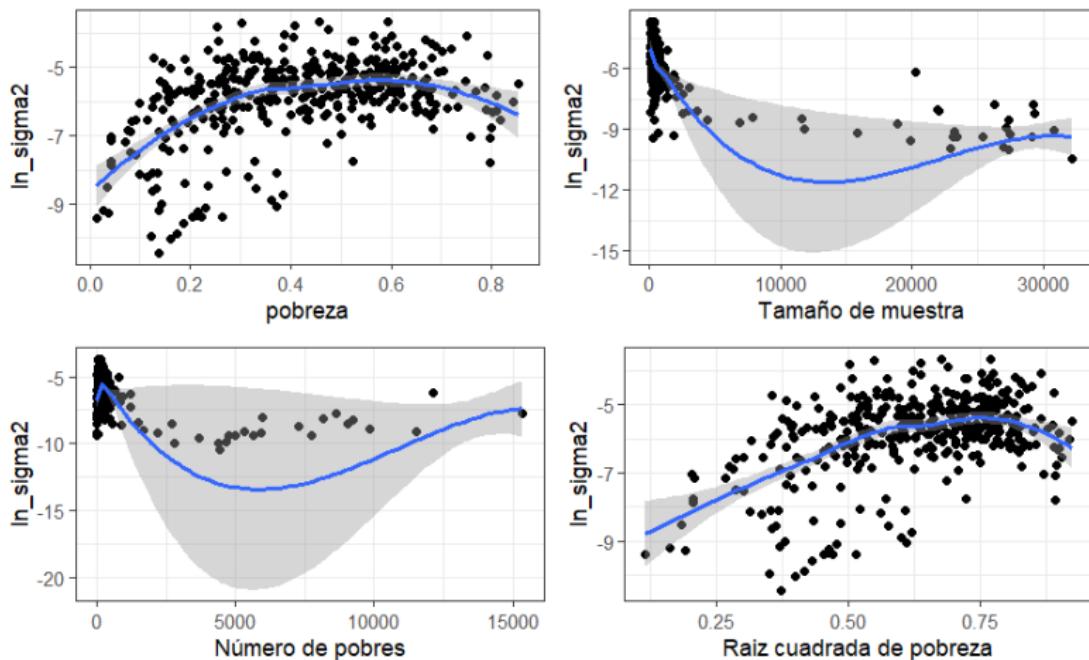


Figure 4: Diagramas de dispersión

Modelo para la varianza

El modelo definido para el conjunto de datos es el siguiente.

$$\log(\hat{\sigma}^2) = \hat{\theta}_{dir} + n_d^2 + \sqrt{\hat{\theta}_{dir}}$$

El resultado del modelo se muestra a continuación:

Table 4: Resumen del modelo

Characteristic	Beta	95% CI	p-value
pobreza	-12	-14, -9.5	<0.001
I(nd^2)	0.00	0.00, 0.00	<0.001
I(sqrt(pobreza))	16	14, 19	<0.001
R ²	0.608		
Adjusted R ²	0.604		

Estimación para Δ y predicción.

A partir de la estimación del modelo se debe obtener el valor de la constante Δ para lo cual se usa el siguiente código.

```
delta.hat = sum(baseFGV$vardir) /  
sum(exp(fitted.values(FGV1)))
```

Por último se tiene la varianza suavizada.

```
hat.sigma <-  
data.frame(  
  dam2 = baseFGV$dam2,  
  hat_var = delta.hat * exp(fitted.values(FGV1)))
```

Validación de resultados.

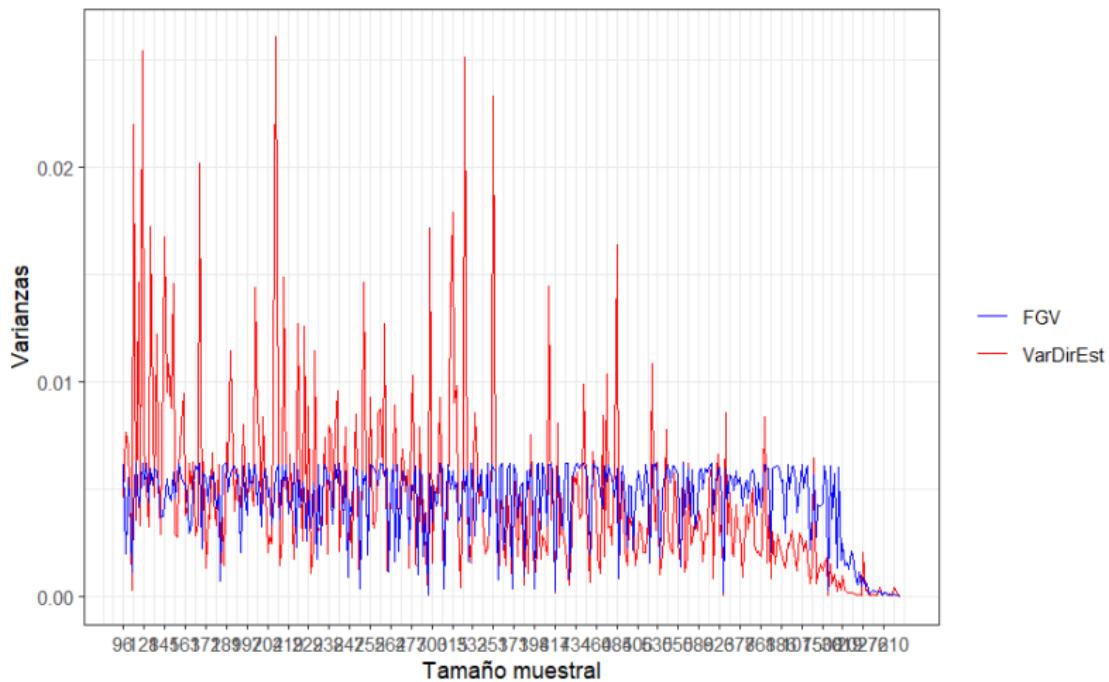


Figure 5: FGV y Varianza directa, por tamaño de muestra

Modelos de área.

Modelo de Fay Herriot

- ▶ El Modelo de Fay Herriot, propuesto por Fay y Herriot en 1979, es ampliamente utilizado en la estimación de áreas pequeñas. Este enfoque estadístico se aplica cuando la información a nivel de individuo es limitada, pero se dispone de datos a nivel de áreas y de información auxiliar relacionada con estos datos.
- ▶ El modelo establece una relación entre los indicadores de las áreas, θ_d , que varían en función de un vector de covariables x_d . Se formula como $\theta_d = x_d^T \beta + u_d$, donde u_d es un efecto aleatorio específico para cada área.

Modelo de Fay Herriot

- Dado que los valores reales de los indicadores θ_d no son observables, se utiliza el estimador directo $\hat{\theta}_d^{DIR}$ para estimarlos, lo que introduce un error de muestreo. Es decir,

$$\hat{\theta}_d^{DIR} = \theta + e_d$$

- El modelo se ajusta teniendo en cuenta el error de muestreo e_d , y las varianzas $\sigma_{e_d}^2$ se estiman a partir de los microdatos de la encuesta. Esto es:

$$\hat{\theta}_d^{DIR} = x_d^T \beta + u_d + e_d$$

Modelo de Fay Herriot

El mejor predictor lineal insesgado (BLUP) bajo el modelo Fay Herriot se calcula como $\tilde{\theta}_d^{FH}$, y se basa en el uso de γ_d para ponderar adecuadamente el estimador directo y la información auxiliar, permitiendo una estimación más precisa de los indicadores en áreas pequeñas. Su ecuación esta dada por:

$$\tilde{\theta}_d^{FH} = x^T d \tilde{\beta} + \tilde{u}_d$$

,

donde $\tilde{u}_d = \gamma_d (\hat{\theta}_d^{DIR} - x_d^T \tilde{\beta})$ y $\gamma_d = \frac{\sigma_u^2}{\sigma_u^2 + \sigma_{e_d}^2}$.

Modelo de área para la estimación de la pobreza

Sea P_d la probabilidad de encontrar una persona en condición de pobreza en el d -ésimo dominio de la población. Entonces, el estimador directo de P_d se puede escribir como:

$$\hat{P}_d^{DIR} = P_d + e_d$$

Ahora bien, P_d se puede modelar de la siguiente manera,

$$P_d = x_d^T \beta + u_d$$

Modelo de área para la estimación de la pobreza

Reescribiendo \hat{P}_d^{DIR} en términos de las dos ecuaciones anteriores tenemos:

$$\hat{P}_d^{DIR} = x_d^T \beta + u_d + e_d$$

Ahora, es posible suponer que:

- $\hat{P}_d^{DIR} \sim N(x_d^T \beta, \sigma_u^2 + \sigma_{e_d}^2)$,
- $\hat{P}_d^{DIR} | u_d \sim N(x_d^T \beta + u_d, \sigma_{e_d}^2)$ y
- $u_d \sim N(0, \sigma_u^2)$

Distribuciones previas.

Las distribuciones previas para β y σ_u^2

$$\beta_p \sim N(0, 10000)$$

$$\sigma_u^2 \sim IG(0.0001, 0.0001)$$

por tanto, el estimador bayesiano para P_d esta dado como
 $\tilde{P}_d = E(P_d | \hat{P}_d^{DIR})$

Procedimiento de estimación de la pobreza en los municipios de colombia

Las covariables disponibles se muestran en la siguiente tabla, estas fueron obtenidas previamente.

Table 5: Covariables disponibles

dam	dam2	area1	sexo2	edad2	edad3	edad4	edad5
05	05001	0.9832	0.5299	0.2671	0.2201	0.2355	0.1060
05	05002	0.3953	0.4807	0.2229	0.1977	0.2497	0.1281
05	05004	0.3279	0.4576	0.2376	0.2075	0.2316	0.1218
05	05021	0.5770	0.5020	0.2191	0.1946	0.2357	0.1274
05	05030	0.4859	0.5063	0.2571	0.2047	0.2507	0.0997

Modelo de FH: Rutina en STAN

```
data {  
    int<lower=0> N1; // number of data items  
    int<lower=0> N2; // number of data items for prediction  
    int<lower=0> p; // number of predictors  
    matrix[N1, p] X; // predictor matrix  
    matrix[N2, p] Xs; // predictor matrix  
    vector[N1] y; // predictor matrix  
    vector[N1] sigma_e; // known variances  
}  
  
parameters {  
    vector[p] beta; // coefficients for predictors  
    real<lower=0> sigma2_u;  
    vector[N1] u;  
}
```

Modelo de FH: Rutina en STAN

```
transformed parameters{
  vector[N1] theta;
  vector[N1] thetaSyn;
  vector[N1] thetaFH;
  vector[N1] gammaj;
  real<lower=0> sigma_u;
  thetaSyn = X * beta;
  theta = thetaSyn + u;
  sigma_u = sqrt(sigma2_u);
  gammaj = to_vector(sigma_u ./ (sigma_u + sigma_e));
  thetaFH = (gammaj) .* y + (1-gammaj).*thetaSyn;
}
```

Modelo de FH: Rutina en STAN

```
model {  
    // likelihood  
    y ~ normal(theta, sigma_e);  
    // priors  
    beta ~ normal(0, 100);  
    u ~ normal(0, sigma_u);  
    sigma2_u ~ inv_gamma(0.0001, 0.0001);  
}  
  
generated quantities{  
    vector[N2] y_pred;  
    for(j in 1:N2) {  
        y_pred[j] = normal_rng(Xs[j] * beta, sigma_u);  
    }  
}
```

Preparando los insumos para STAN

- Definir el modelo de área

```
formula_mod <- formula(  
~ sexo2 + anoest2 + anoest3 +  
anoest4 + edad2 + edad3 + edad4 + edad5 + etnia1 +  
etnia2 + tasa_desocupacion + luces_nocturnas +  
cubrimiento_cultivo + alfabeto  
)
```

Preparando los insumos para STAN

- ▶ Dividir la base de datos en dominios observados y no observados.

```
# Dominios observados.  
data_dir <- base_FH %>% filter(!is.na(pobreza))  
  
Xdat <- model.matrix(formula_mod, data = data_dir)  
  
# Dominios NO observados.  
data_syn <-  
  base_FH %>% anti_join(data_dir %>% select(dam2))  
  
Xs <- model.matrix(formula_mod, data = data_syn)
```

Preparando los insumos para STAN

► Creando lista de parámetros para STAN

```
sample_data <- list(  
  N1 = nrow(Xdat),    # Observados.  
  N2 = nrow(Xs),     # NO Observados.  
  p  = ncol(Xdat),    # Número de regresores.  
  X  = as.matrix(Xdat), # Covariables Observados.  
  Xs = as.matrix(Xs),   # Covariables NO Observados  
  y  = as.numeric(data_dir$pobreza), # Estimación directa  
  sigma_e = sqrt(data_dir$hat_var)   # Error de estimación  
)
```

Compilando el modelo en STAN

La forma de compilar el código de STAN desde R.

```
library(rstan)
fit_FH_normal <- "www/02_FH_Nornal/17FH_normal.stan"
options(mc.cores = parallel::detectCores())
model_FH_normal <- stan(
  file = fit_FH_normal,
  data = sample_data,
  verbose = FALSE,
  warmup = 500,
  iter = 1000,
  cores = 4
)
saveRDS(object = model_FH_normal,
        file = "www/02_FH_Nornal/model_FH_normal.rds")
```

Resultados del modelo para los dominios observados.

Empleando la función `ppc_dens_overlay()` para graficar una comparación entre la distribución empírica de la variable observada pobreza en los datos y las distribuciones predictivas posteriores simuladas para la misma variable.

```
y_pred_B <- as.array(model_FH_normal,
                      pars = "theta") %>%
  as_draws_matrix()

rowsrandom <- sample(nrow(y_pred_B), 100)

y_pred2 <- y_pred_B[rowsrandom,]

ppc_dens_overlay(y = as.numeric(data_dir$pobreza),
                  y_pred2)
```

Chequeo Predictivo Posterior

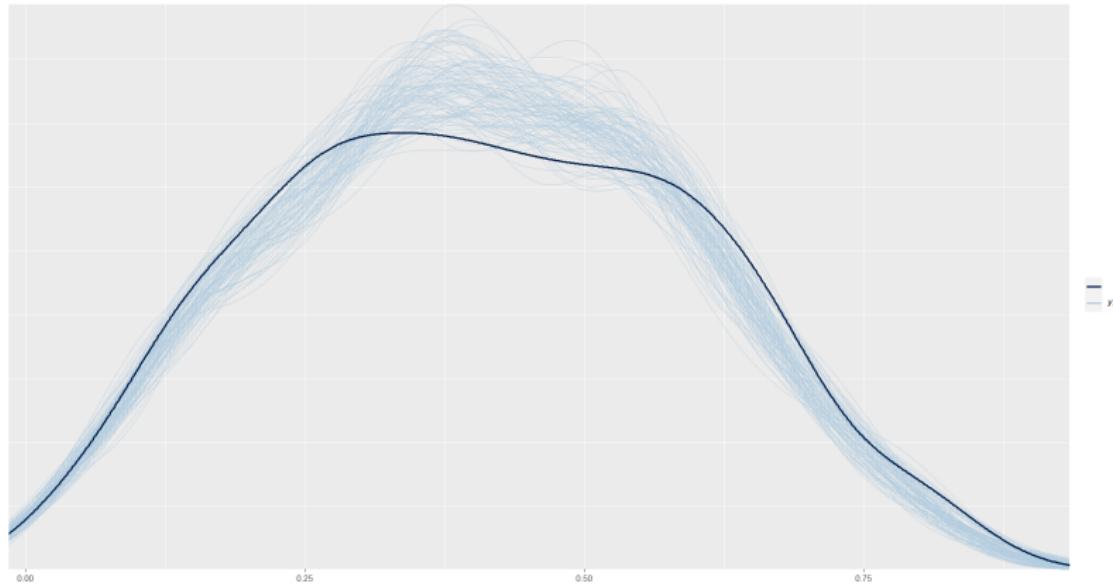


Figure 6: PPC

Validacion de convergencia de cadenas σ^2

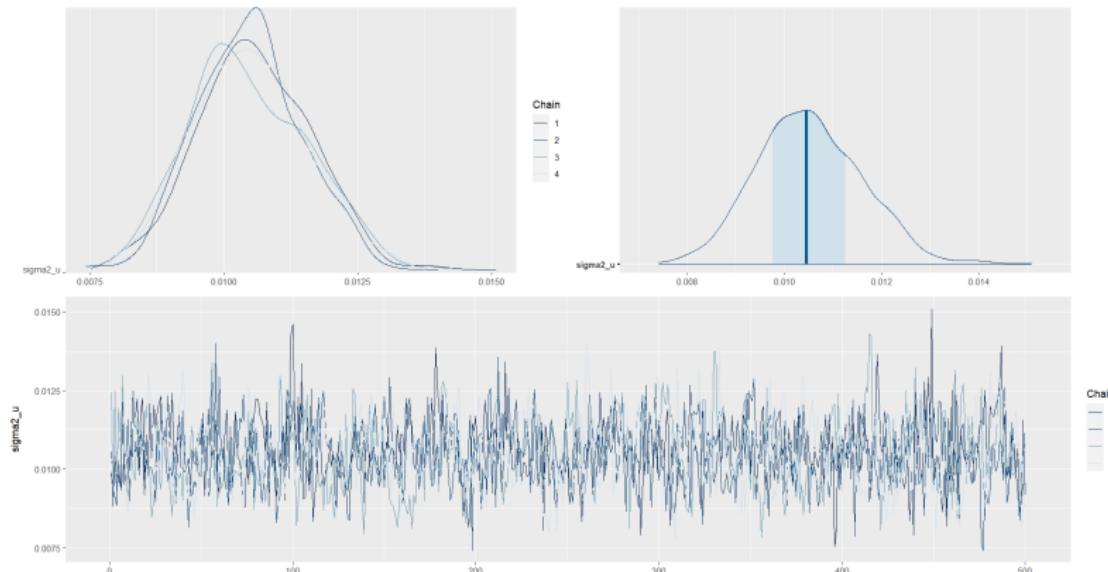
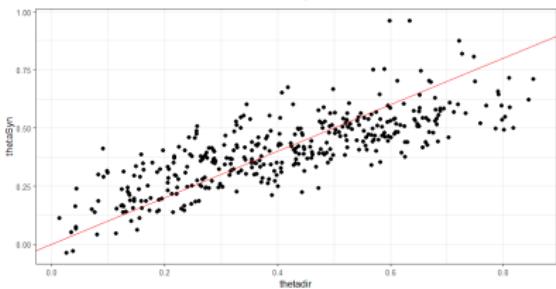
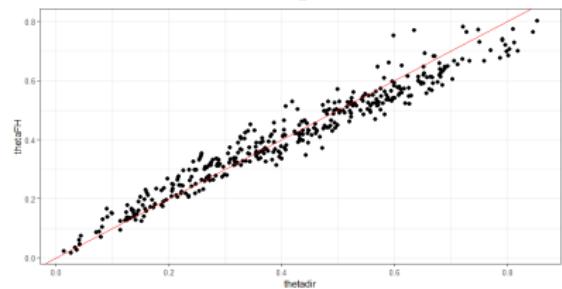
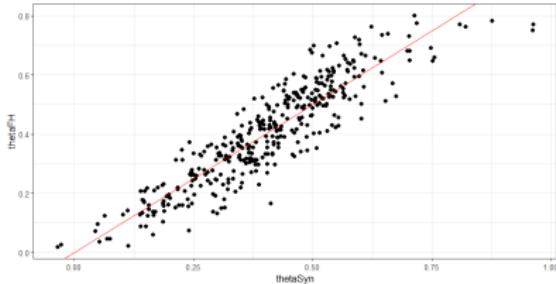
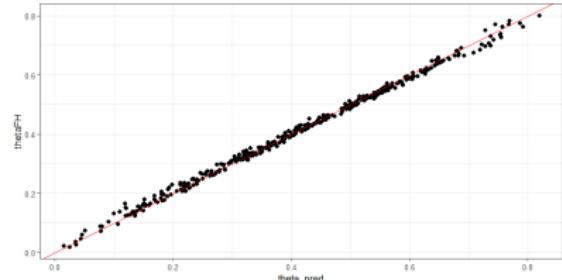


Figure 7: Convergencia de la cadena

Comparación de las estimaciones



Proceso de Benchmarking

- Del censo extraer el total de personas por DAM2

dam	dam2	total_pp	dam_pp
05	05001	2372330	44164417
05	05002	17599	44164417
05	05004	2159	44164417
05	05021	3839	44164417
05	05030	26821	44164417
05	05031	20265	44164417
05	05034	38144	44164417
05	05036	5027	44164417
05	05038	10500	44164417
05	05040	14502	44164417

Estimación directa

Obtener las estimaciones directa por DAM o el nivel de agregación en el cual la encuesta es representativa.

```
directoDam <- diseno %>%
  group_by(Agregado = "Nacional") %>%
  summarise(
    theta_dir = survey_mean(pobreza, vartype = c("ci"))
  )
```

Agregado	theta_dir	theta_dir_low	theta_dir_upp
Nacional	0.2986	0.2935	0.3038

Calculo de ponderadores

Luego de organizar la información anterior se realiza el calculo de los pesos para el Benchmark

```
estimacionesPre <-
  readRDS("www/02_FH_Nornal/05_tabla_estimacionesPre.rds")
temp <- estimacionesPre %>%
  inner_join(N_dam_pp) %>%
  mutate(theta_dir = directoDam$theta_dir)
R_dam2 <- temp %>%
  summarise(
    R_dam_RB = unique(theta_dir) /
      sum((total_pp / dam_pp) * theta_pred))
```

$$\begin{array}{c} \hline \text{R_dam_RB} \\ \hline 1.016 \\ \hline \end{array}$$

Estimación con el modelo de área despues del Benchmarking

```
pesos <- temp %>%
  mutate(W_i = total_pp / dam_pp) %>%
  select(dam2, W_i)

estimacionesBench <- estimacionesPre %>%
  mutate(R_dam_RB = R_dam2$R_dam_RB) %>%
  mutate(theta_pred_RBench = R_dam_RB * theta_pred) %>%
  select(dam, dam2, theta_pred, theta_pred_RBench)
```

dam	dam2	W_i	theta_pred	theta_pred_RBench
05	05001	0.0537	0.1593	0.1618
05	05002	0.0004	0.4130	0.4194
05	05031	0.0005	0.4121	0.4185

Validación de los resultados.

Este código junta las estimaciones del modelo con pesos de benchmarking con los valores observados y sintéticos, y luego resume las estimaciones combinadas para compararlas con la estimación directa obtenida anteriormente.

```
temp <- estimacionesBench %>%
  left_join(estimacionesPre) %>%
  summarise(
    thetaSyn = sum(W_i * thetaSyn),
    thetaFH = sum(W_i * theta_pred),
    theta_RBench = sum(W_i * theta_pred_RBench)
  ) %>%
  mutate(
    theta_dir = directoDam$theta_dir,
    theta_dir_low = directoDam$theta_dir_low,
    theta_dir_upp = directoDam$theta_dir_upp
  )
```

Resultado de la Validación

Table 6: Comparación de las estimaciones

theta_dir_low	theta_dir_upp	Metodo	Estimacion
0.2935	0.3038	thetaSyn	0.2955
0.2935	0.3038	thetaFH	0.2941
0.2935	0.3038	theta_RBench	0.2986
0.2935	0.3038	theta_dir	0.2986

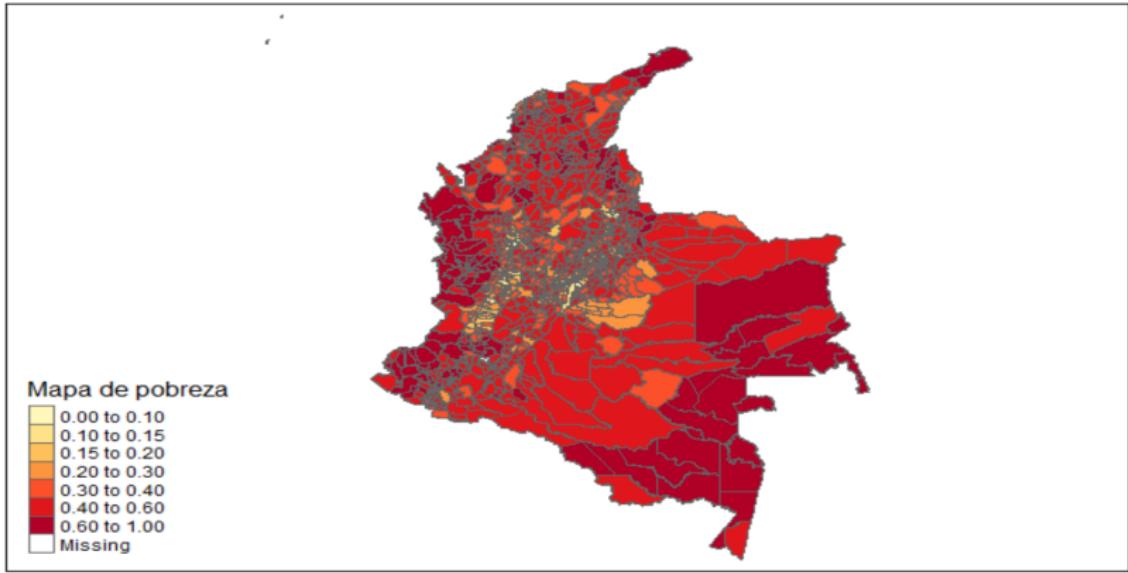


Figure 8: Mapa de pobreza

Modelo de área: Transformación Arcoseno.

- ▶ En el modelo de Fay-Herriot, la combinación lineal de covariables puede generar valores que no están dentro del rango aceptable para una proporción.
- ▶ Para abordar esto, se aplica una transformación arcoseno a los estimadores: $\hat{z}_d = \arcsin\left(\sqrt{\hat{\theta}_d}\right)$.

Varianza de la Transformación Arcoseno

- La varianza de la transformación arcoseno está relacionada con el factor de corrección DEFF y el tamaño de muestra efectivo:

$$Var(\hat{z}_d) = \frac{\widehat{DEFF}_d}{4 \times n_d} = \frac{1}{4 \times n_{d, efectivo}}$$

Especificación del Modelo de Fay-Herriot

- ▶ El modelo de Fay-Herriot se define con una variable latente Z_d que sigue una distribución normal.
- ▶ La media de Z_d (μ_d) se relaciona con las covariables a través de $x_d^T \beta + u_d$.
- ▶ La relación entre la variable latente θ_d y el estimador directo se establece como $\theta_d = (\sin(\mu_d))^2$.

Lo anterior se simplifica como:

1. $Z_d | \mu_d, \sigma_d^2 \sim N(\mu_d, \sigma_d^2)$
2. $\mu_d = x_d^T \beta + u_d$
3. $\theta_d = (\sin(\mu_d))^2$

Distribuciones Previas

Se especifican distribuciones previas para los parámetros del modelo: - $\beta \sim N(0, 1000)$ - $\sigma_u^2 \sim IG(0.0001, 0.0001)$.

Modelo de área: Rutina en STAN

El código es similar al anterior, aquí se muestran las variaciones

```
transformed parameters{
  vector[N1] theta;
  vector[N1] lp;
  real<lower=0> sigma_u;
  lp = X * beta + u;
  sigma_u = sqrt(sigma2_u);
  for(k in 1:N1){
    theta[k] = pow(sin(lp[k]), 2);
  }
}
```

Modelo de FH: Rutina en STAN

```
model {  
    // likelihood  
    y ~ normal(lp, sigma_e);  
    // priors  
    beta ~ normal(0, 100);  
    u ~ normal(0, sigma_u);  
    sigma2_u ~ inv_gamma(0.0001, 0.0001);  
}
```

Procedimiento de estimación

Para la base preparada previamente hay que seleccionar y transformar las columnas de interés.

```
statelevel_predictors_df <-
  readRDS("www/03_FH_Arcsin/statelevel_predictors.rds")
base_FH <-
  readRDS("www/03_FH_Arcsin/base_FH_2018.rds") %>%
  transmute(
    dam2,          # id dominios
    pobreza,
    T_pobreza = asin(sqrt(pobreza)), # creando zd
    n_effec = n_eff_FGV,           # n efectivo
    varhat = 1/(4*n_effec)        # varianza para zd
  )
base_FH <- full_join(base_FH,
                      statelevel_predictors_df, by = "dam2" )
```

Preparando los insumos para STAN

Selección de las covariables, que corresponden a las seleccionadas previamente.

```
names_cov <- c(  
  "sexo2" , "anoest2" , "anoest3",   "anoest4",  
  "edad2" , "edad3" , "edad4" , "edad5" , "etnia1",  
  "etnia2" , "tasa_desocupacion" , "luces_nocturnas" ,  
  "cubrimiento_cultivo" , "alfabeta"  
)
```

Dividir el set de datos.

El proceso de estimación y predicción se hace por separado dentro de STAN

- ▶ Dominios observados.

```
data_dir <- base_FH %>% filter(!is.na(T_pobreza))
Xdat <- cbind(inter = 1,data_dir[,names_cov])
```

- ▶ Dominios NO observados.

```
data_syn <-
  base_FH %>% anti_join(data_dir %>% select(dam2))
Xs <- cbind(inter = 1,data_syn[,names_cov])
```

Lista de parámetros para STAN

El motor de procesamiento de STAN se basa en C++, por lo que hace necesario que los argumentos para ejecutar los códigos ingresen en forma de lista.

```
sample_data <- list(  
  N1 = nrow(Xdat),           # Observados.  
  N2 = nrow(Xs),             # NO Observados.  
  p = ncol(Xdat),            # Número de regresores.  
  X = as.matrix(Xdat),       # Covariables Observados.  
  Xs = as.matrix(Xs),         # Covariables NO Observados  
  y = as.numeric(data_dir$T_pobreza),  
  sigma_e = sqrt(data_dir$varhat)  
)
```

Compilando el modelo en STAN

```
fit_FH_arco seno <-
  "www/03_FH_Arcsin/15FH_arcsin_normal.stan"

model_FH_arco seno <- stan(
  file = fit_FH_arco seno,
  data = sample_data,
  verbose = FALSE,
  warmup = 500,
  iter = 1000,
  cores = 4
)
saveRDS(model_FH_arco seno,
        "www/03_FH_Arcsin/model_FH_arco seno.rds")
```

Resultados del modelo para los dominios observados.

De forma similar al modelo de Fay Herriot se realiza el gráfico con el chequeo predictivo posterior.

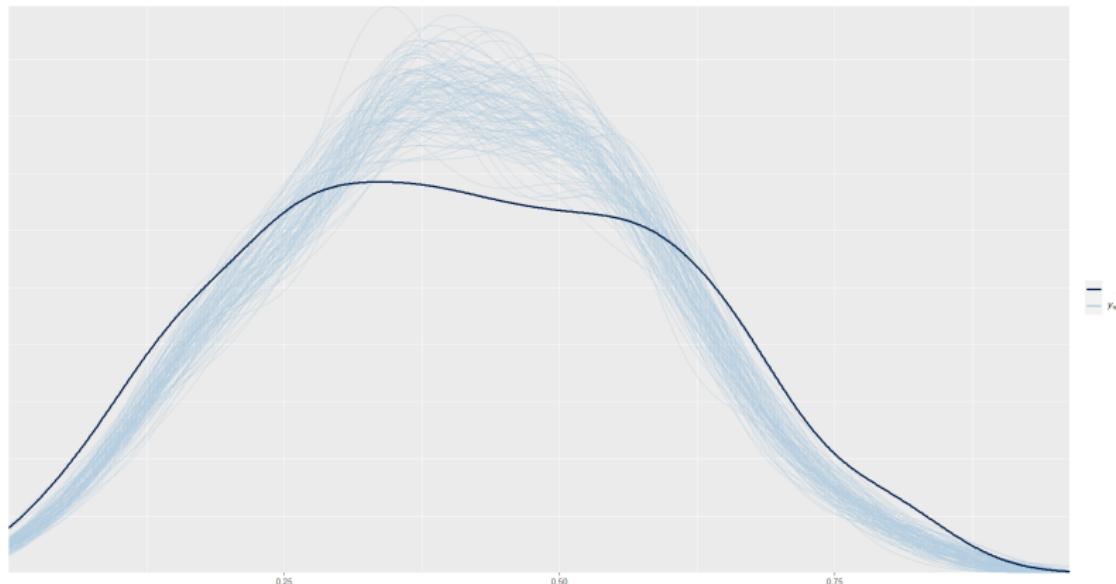


Figure 9: PPC Arcosin

Análisis gráfico de la convergencia de las cadenas de σ_u^2 .

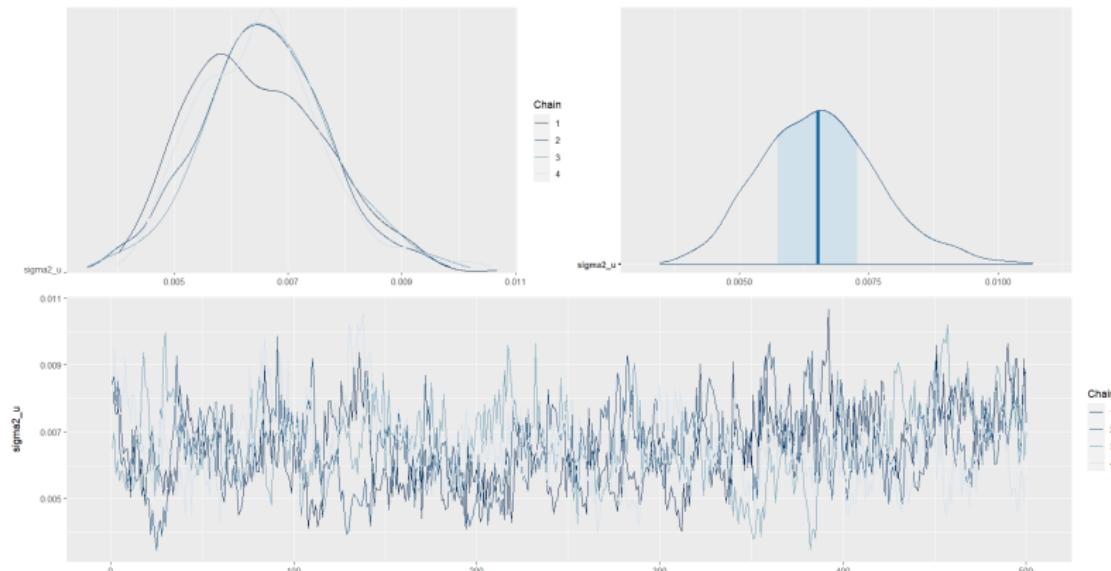


Figure 10: Recorrido de las cadenas

Mapa de pobreza con transformación Arcosin

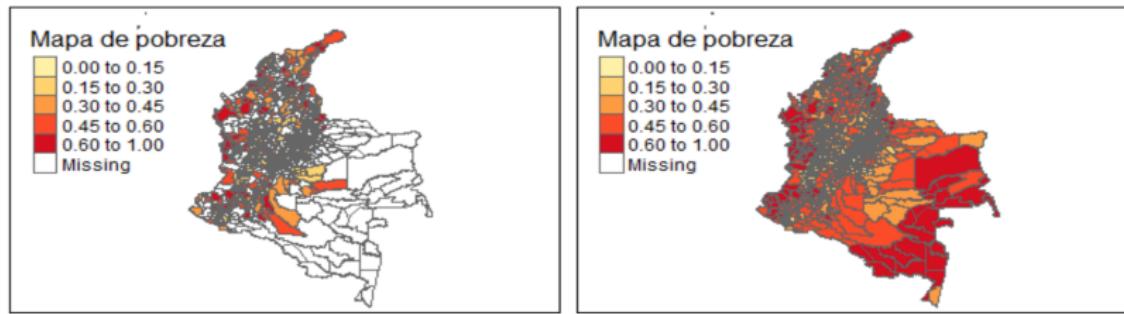


Figure 11: Mapa de pobreza con transformación Arcosin

Mapa de los coeficientes de variación para la pobreza

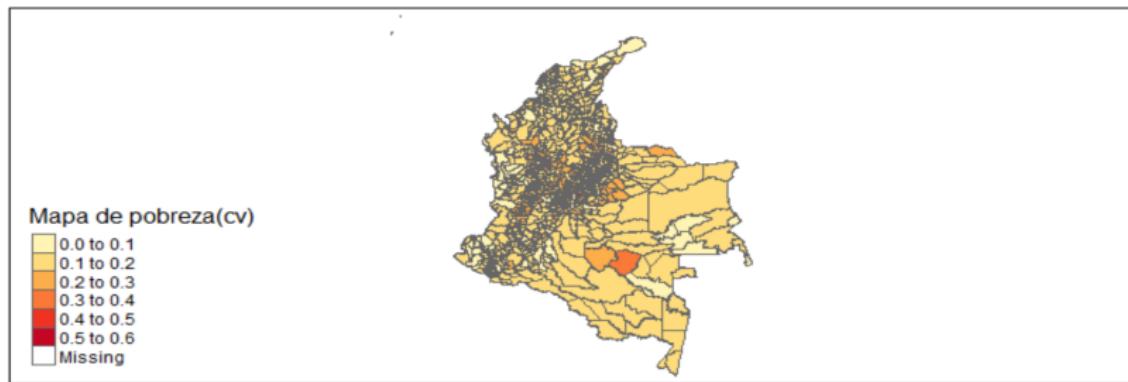


Figure 12: Mapa de los coeficientes de variación

Modelos de área con variable respuesta Beta.

- ▶ El modelo beta-logístico se introdujo inicialmente en el contexto de un enfoque de Estimación de Mejor Predicción (EBP) por Jiang y Lahiri en 2006. Fue utilizado para estimar medias de dominio en poblaciones finitas.
- ▶ El modelo área beta-logístico se define a través de la siguiente expresión:
 - ▶ $\hat{p}_d \mid P_d \sim \text{beta}(a_d, b_d)$.
 - ▶ La función de enlace se relaciona con los parámetros del modelo:
 - ▶ $\text{logit}(P_d) \mid \beta, \sigma_u^2 \sim N(x_d^T \beta, \sigma_u^2)$.

Estimación de Parámetros

- ▶ Los parámetros a_d y b_d se estiman de la siguiente manera:
 - ▶ $a_d = P_d \times \phi_d$
 - ▶ $b_d = (1 - P_d) \times \phi_d$
- ▶ Donde $\phi_d = \frac{n_d}{\widehat{DEF}_d} - 1 = n_{d, efectivo} - 1$.
- ▶ Se especifican distribuciones previas para los parámetros del modelo:
 - ▶ $\beta_k \sim N(0, 10000)$
 - ▶ $\sigma_u^2 \sim IG(0.0001, 0.0001)$.

Modelo de área: Rutina en STAN

En este bloque de código vemos la transformación que se realiza sobre los parámetros de entrada.

```
transformed parameters{
  vector[N1] LP;
  real<lower=0> sigma_u;
  vector[N1] theta;
  LP = X * beta + u;
  sigma_u = sqrt(sigma2_u);
  for (i in 1:N1) {
    theta[i] = inv_logit(LP[i]);
  }
}
```

Modelo de FH: Rutina en STAN

```
model {  
    // model calculations  
    vector[N1] a;  
    vector[N1] b;  
  
    for (i in 1:N1) {  
        a[i] = theta[i] * phi[i];  
        b[i] = (1 - theta[i]) * phi[i];  
    }  
  
    // priors  
    beta ~ normal(0, 100);  
    u ~ normal(0, sigma_u);  
    sigma2_u ~ inv_gamma(0.0001, 0.0001);  
  
    // likelihood  
    y ~ beta(a, b);  
}
```

Procedimiento de estimación

En forma similar a los modelos anteriores hacemos uso de la base previamente preparada

```
base_FH <-  
readRDS("www/04_FH_Beta_y_Binomial/base_FH_2018.rds") %>%  
  select(dam2, pobreza, n_eff_FGV)  
  
base_FH <- full_join(base_FH,  
                      statelevel_predictors_df, by = "dam2")
```

Las covariables son las mismas que se emplearon en los modelos anteriores.

Dividir el set de datos.

El proceso de estimación y predicción se hace por separado dentro de STAN

- ▶ Dominios observados.

```
data_dir <- base_FH %>% filter(!is.na(T_pobreza))
Xdat <- cbind(inter = 1,data_dir[,names_cov])
```

- ▶ Dominios NO observados.

```
data_syn <-
  base_FH %>% anti_join(data_dir %>% select(dam2))
Xs <- cbind(inter = 1,data_syn[,names_cov])
```

Lista de parámetros para STAN

```
sample_data <- list(  
  N1 = nrow(Xdat),    # Observados.  
  N2 = nrow(Xs),     # NO Observados.  
  p  = ncol(Xdat),      # Número de regresores.  
  X  = as.matrix(Xdat),  # Covariables Observados.  
  Xs = as.matrix(Xs),    # Covariables NO Observados  
  y  = as.numeric(data_dir$pobreza),  
  phi = data_dir$n_eff_FGV - 1  
)
```

Compilando el modelo en STAN

```
fit_FH_beta_logitic <-
"www/04_FH_Beta_y_Binomial/16FH_beta_logitc.stan"

model_FH_beta_logitic <- stan(
  file = fit_FH_beta_logitic,
  data = sample_data,
  verbose = FALSE,
  warmup = 500,
  iter = 1000,
  cores = 4
)
saveRDS(model_FH_beta_logitic,
file = "www/04_FH_Beta_y_Binomial/model_FH_beta.rds")
```

Resultados del modelo para los dominios observados.

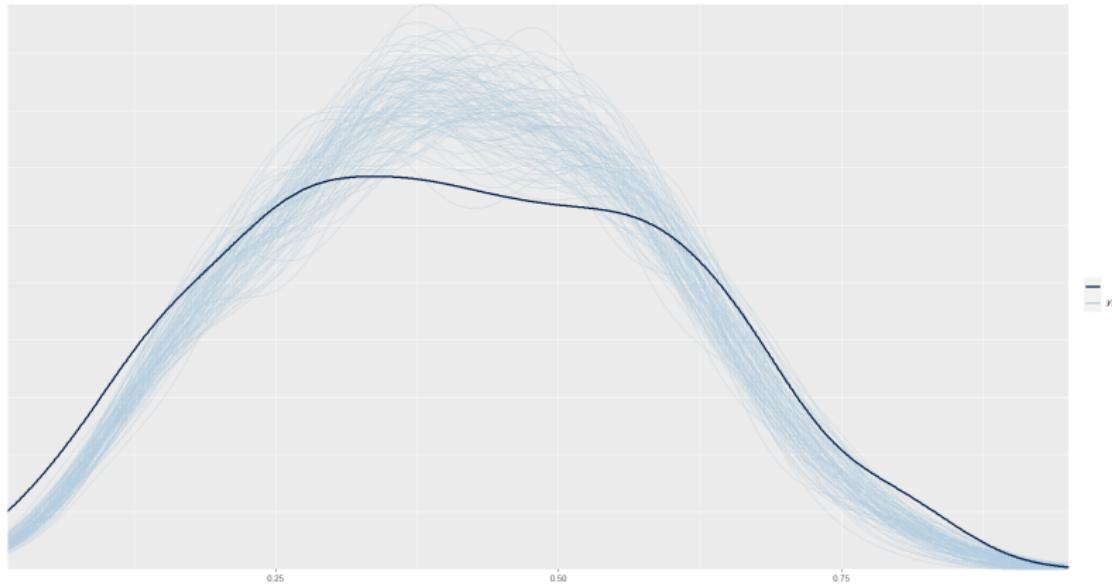


Figure 13: PPC modelo de área Beta

Análisis gráfico de la convergencia de las cadenas de σ_u^2 .

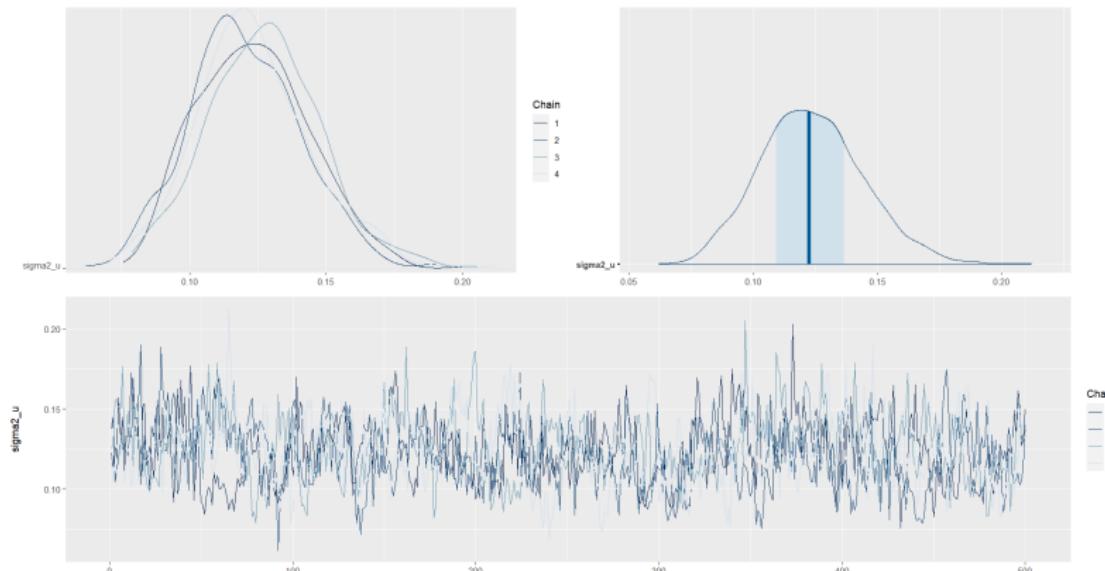


Figure 14: Recorrido de las cadenas

Mapa de pobreza con modelo de área de respuesta beta.

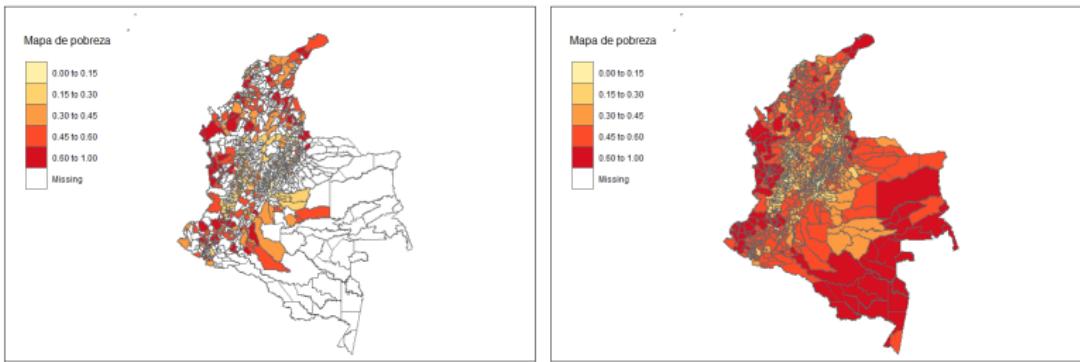


Figure 15: Mapa de pobreza con el modelo de área de respuesta beta.

Mapa de los coeficientes de variación para la pobreza

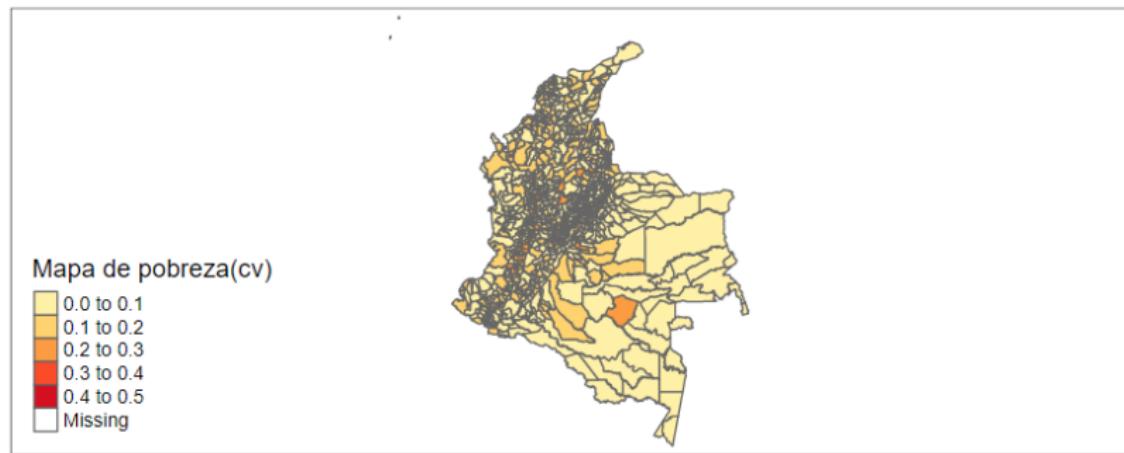


Figure 16: Mapa de los coeficientes de variación

Modelos de área con variable respuesta Binomial.

- ▶ El modelo de área de Fay-Herriot puede ser sustituido por un Modelo Mixto Lineal Generalizado (GLMM) cuando los datos observados son inherentemente discretos, como recuentos de personas u hogares con ciertas características.
- ▶ En un GLMM, se asume una distribución binomial para los datos Y_d con probabilidad de éxito θ_d y un modelo logístico para θ_d con errores normales en la escala logit.

Ecuación del modelo

El modelo se formula de la siguiente manera:

- ▶ $Y_d | \theta_d, n_d \sim Binomial(n_d, \theta_d)$
- ▶ $logit(\theta_d) = \log\left(\frac{\theta_d}{1-\theta_d}\right) = x_d^T \beta + u_d$

donde $u_d \sim N(0, \sigma_u^2)$ y n_d es el tamaño de la muestra para el área d .

Consideraciones para el modelo

Para muestras complejas, surgen dos problemas:

- ▶ Los valores de Y_d no son enteros y se ven afectados por las ponderaciones de la encuesta.
- ▶ La varianza muestral en la distribución binomial no es precisa.

Propuesta de Carolina Franco.

- ▶ Se introduce un **tamaño de muestra efectivo** \tilde{n}_d y un **número de muestra efectivo de éxitos** \tilde{Y}_d para abordar estos problemas y mantener la estimación directa de la pobreza y su varianza correspondiente.
- ▶ Dado lo anterior, es posible suponer que

$$\tilde{n}_d \sim \frac{\check{\theta}_d (1 - \check{\theta}_d)}{\widehat{Var}(\hat{\theta}_d)}$$

con $\check{\theta}_d$ es una preliminar predicción basada en el modelo para la proporción poblacional, $\hat{\theta}_i$ la estimación directa y $\widehat{Var}(\hat{\theta}_d)$ la estimación de la varianza de muestreo.

- ▶ Luego, se asume que \tilde{n}_d es proporcional a la varianza ajustada y que $\tilde{Y}_d = \tilde{n}_d \times \hat{\theta}_d$.

Distribuciones previas

- Se especifican las distribuciones previas para los parámetros β y σ_u^2 :
 - $\beta \sim N(0, 10000)$
 - $\sigma_u^2 \sim IG(0.0001, 0.0001)$

Modelo de área: Rutina en STAN

En este bloque de código vemos la transformación que se realiza sobre los parámetros de entrada.

```
transformed parameters {
    vector[N1] LP;
    vector[N1] theta;
    real<lower=0> sigma_u;

    sigma_u = sqrt(sigma2_u);
    LP = X * beta + u;
    theta = inv_logit(LP);
}
```

Modelo de FH: Rutina en STAN

```
model {  
    to_vector(beta) ~ normal(0, 10000);  
    u ~ normal(0, sigma_u);  
    sigma2_u ~ cauchy(0, 1000);  
    for(ii in 1:N1){  
        y_effect[ii] ~ binomial(n_effec[ii], theta[ii]);  
    }  
}  
  
generated quantities {  
    real ypred[N2];  
    vector[N2] thetaLP;  
    vector[N2] LP_pred;  
    LP_pred = Xs * beta;  
    thetaLP = inv_logit(LP_pred);  
}
```

Procedimiento de estimación

Lectura de la base de datos con las estimaciones directas.

```
base_FH <-  
readRDS("www/04_FH_Beta_y_Binomial/base_FH_2018.rds") %>%  
  select(dam2, pobreza, n_eff_FGV)  
  
base_FH <- full_join(base_FH,  
                      statelevel_predictors_df, by = "dam2")
```

Las covariables son las mismas que se emplearon en los modelos anteriores.

Dividir el set de datos.

El proceso de estimación y predicción se hace por separado dentro de STAN

- ▶ Dominios observados.

```
data_dir <- base_FH %>% filter(!is.na(T_pobreza))
Xdat <- cbind(inter = 1,data_dir[,names_cov])
```

- ▶ Dominios NO observados.

```
data_syn <-
  base_FH %>% anti_join(data_dir %>% select(dam2))
Xs <- cbind(inter = 1,data_syn[,names_cov])
```

Obteniendo parámetros adicionales.

- Tamaño de muestra efectivo \tilde{n}_d

```
n_effec = round(data_dir$n_eff_FGV)
```

- Número de muestra efectivo de éxitos \tilde{Y}_d

```
y_effect = round((data_dir$pobreza)*n_effec)
```

Lista de parámetros para STAN

```
sample_data <- list(  
  N1 = nrow(Xdat),    # Observados.  
  N2 = nrow(Xs),     # NO Observados.  
  p  = ncol(Xdat),      # Número de regresores.  
  X  = as.matrix(Xdat),  # Covariables Observados.  
  Xs = as.matrix(Xs),    # Covariables NO Observados  
  n_effec = n_effec,  
  y_effect  = y_effect  # Estimación directa.  
)
```

Compilando el modelo en STAN

```
fit_FH_binomial <-
"www/04_FH_Beta_y_Binomial/14FH_binomial.stan"

model_FH_Binomial <- stan(
  file = fit_FH_binomial,
  data = sample_data,
  verbose = FALSE,
  warmup = 500,
  iter = 1000,
  cores = 4
)

saveRDS(model_FH_Binomial,
file = "www/04_FH_Beta_y_Binomial/model_FH_Binomial.rds")
```

Resultados del modelo para los dominios observados.

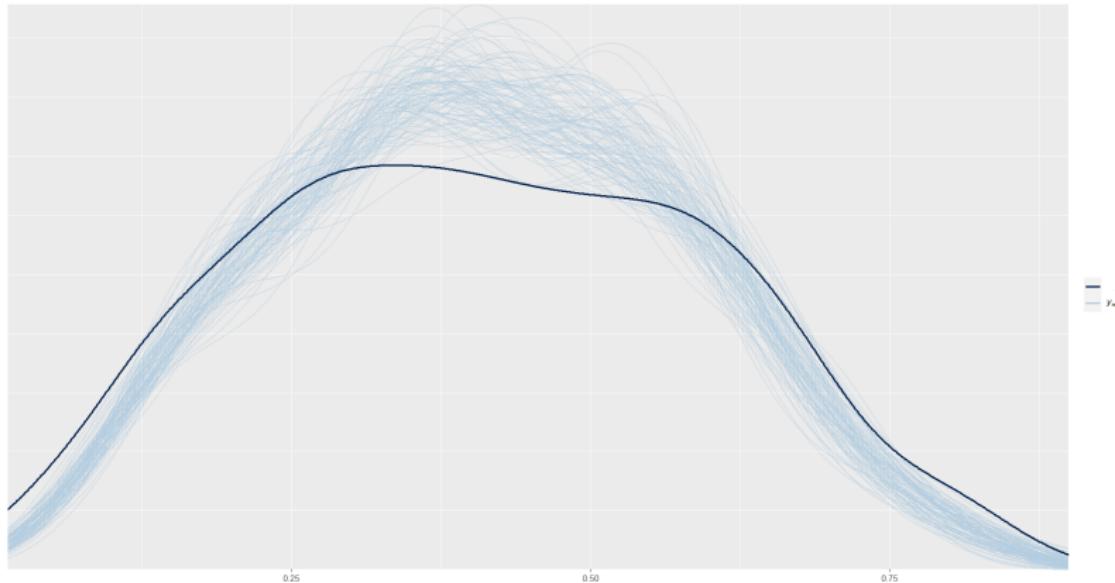


Figure 17: PPC modelo de área Binomial

Análisis gráfico de la convergencia de las cadenas de σ_u^2 .

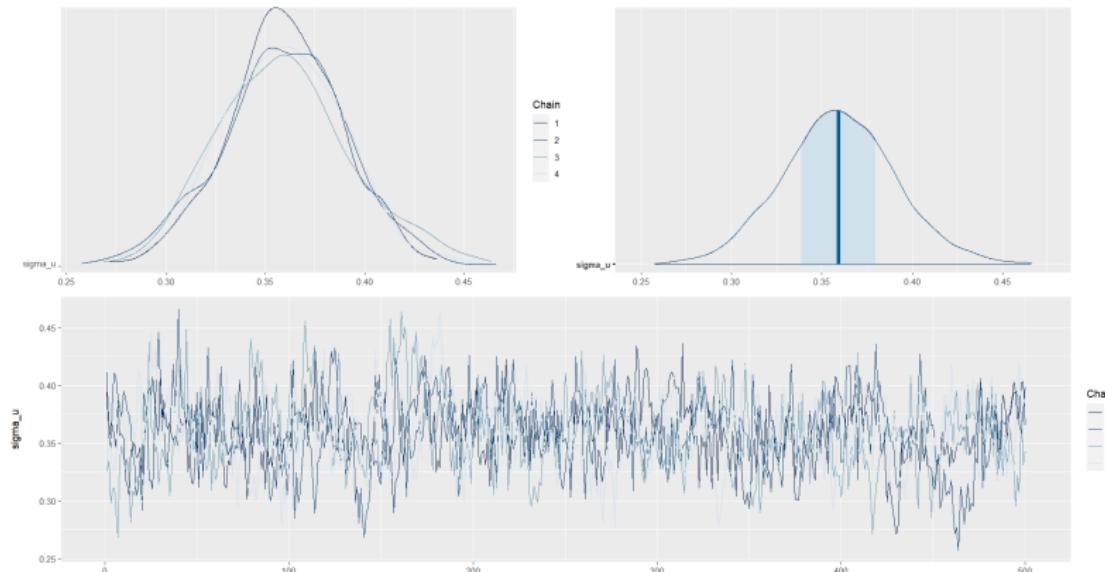


Figure 18: Recorrido de las cadenas

Mapa de pobreza con modelo de área de respuesta binomial

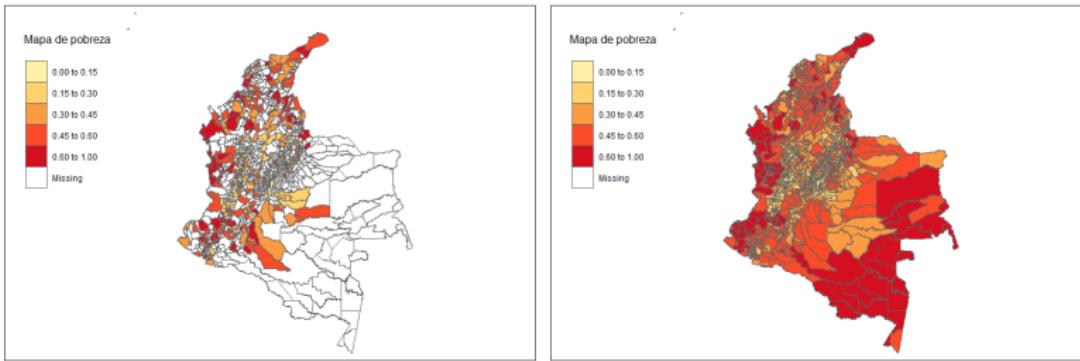


Figure 19: Mapa de pobreza con el modelo de área de respuesta beta.

Mapa de los coeficientes de variación para la pobreza

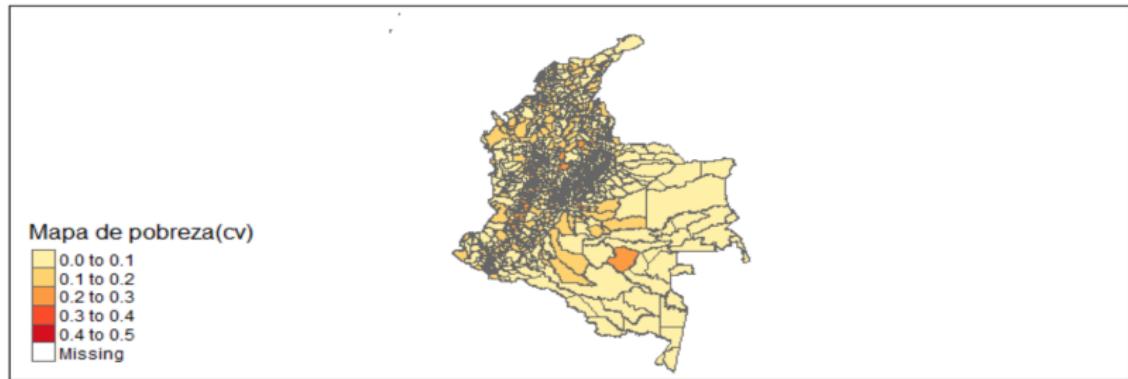


Figure 20: Mapa de los coeficientes de variación

Modelos de unidad.

Modelo de unidad para la estimación del ingreso medio

- ▶ Esta metodología, conocida como “pseudo-EBP,” es un modelo con errores anidados que incorpora los factores de expansión de la encuesta. Este modelo se basa en el concepto del mejor predictor empírico, incorporando información de los microdatos del censo de población.
- ▶ A diferencia de otros modelos con errores anidados de Battese, Harter y Fuller (BHF), no requiere conocer o estimar previamente la varianza de los residuos del modelo. Esto hace que la metodología sea más accesible y práctica.

Mayor Nivel de Desagregación en las Estimaciones

- ▶ Bajo ciertas condiciones, estos modelos permiten una mayor desagregación de las estimaciones. Esto significa que podemos generar estimaciones a nivel de municipio, provincia o comuna, desglosadas por diversas características, como autorreconocimiento étnico, grupo de edad, sexo, discapacidad, entre otros, si se cuenta con covariables a nivel de individuo.

Metodo de estimación del modelo de unidad

Para estimar el ingreso medio de las personas, es decir,

$$\bar{Y}_d = \frac{\sum_{U_d} y_{di}}{N_d}$$

donde y_{di} es el ingreso de cada personas. Note que,

$$\bar{Y}_d = \frac{\sum_{s_d} y_{di} + \sum_{s_d^c} y_{di}}{N_d}$$

Predictión del modelo de unidad

El estimador de \bar{Y} esta dado por:

$$\hat{\bar{Y}}_d = \frac{\sum_{s_d} y_{di} + \sum_{s_d^c} \hat{y}_{di}}{N_d}$$

donde

$$\hat{y}_{di} = E_{\mathcal{M}}(y_{di} | x_d, \beta)$$

'
donde \mathcal{M} hace referencia a la medida de probabilidad inducida por el modelamiento. Así tse tiene que:

$$\hat{\bar{Y}}_d = \frac{\sum_{U_d} \hat{y}_{di}}{N_d}$$

Definición del modelo de unidad.

- ▶ Estamos aplicando un modelo bayesiano para predecir el ingreso medio en áreas no observadas. Esto se basa en la suposición de que los ingresos medios Y_{di} siguen una distribución normal con una media μ_{di} y una varianza σ_e^2 .
- ▶ La media μ_{di} se relaciona con las características individuales X a través de un conjunto de parámetros β , junto con un efecto específico del dominio u_d y un término de error de estimación e_{di} .
- ▶ El modelo:
 - ▶ $Y_{di} \sim N(\mu_{di}, \sigma_e^2)$
 - ▶ $\mu_{di} = x_{di}^T \beta + u_d + e_{di}$

Definición del modelo de unidad.

- ▶ Tanto u_d como e_{di} siguen distribuciones normales, con medias de cero y varianzas σ_u^2 y σ_e^2 respectivamente.
- ▶ Hemos establecido distribuciones previas no informativas para los parámetros β_k y σ_y^2 . Esto significa que asumimos que tenemos poca información previa sobre estos parámetros y, por lo tanto, no les asignamos distribuciones previas específicas.
- ▶ $\beta_k \sim N(0, 1000)$
- ▶ $\sigma_y^2 \sim IG(0.0001, 0.0001)$

Lectura de librerías y funciones de R

- ▶ *plot_interaction*: Esta crea un diagrama de líneas donde se estudia la interacción entre las variables, en el caso de presentar un traslape de las líneas se recomienda incluir el interacción en el modelo.
- ▶ *Aux_Agregado*: Esta es función permite obtener estimaciones a diferentes niveles de agregación, toma mucha relevancia cuando se realiza un proceso repetitivo.

```
library(rstan)
library(rstanarm)
source("www/05_Mod_Ingreso/01_funtings.R")
```

Las funciones están diseñadas específicamente para este proceso

Encuesta de hogares estandarizadas

La base original se recodifica como sigue:

- ▶ Años de estudio (**anoest**) se recodifica como:
 - ▶ 1 → Sin educación
 - ▶ 2 → 1 - 6 años
 - ▶ 3 → 7 - 12 años
 - ▶ 4 → Más de 12
 - ▶ 98 → No aplica
 - ▶ 99 → NS/NR (No sabe/No responde)
- ▶ **Sexo** se recodifica como:
 - ▶ 1 → Hombre
 - ▶ 2 → Mujer
- ▶ Autoreconocimiento (**etnia**) se recodifica como:
 - ▶ 1 → Indígena
 - ▶ 2 → Afrodescendiente
 - ▶ 3 → Otro

Encuesta de hogares estandarizadas

- ▶ **Edad** se recodifica como:
 - ▶ 1 → 0 - 14
 - ▶ 2 → 15 - 29
 - ▶ 3 → 30 - 44
 - ▶ 4 → 45 - 64
 - ▶ 5 → 65 - más
- ▶ Zona urbana/rural (**área**) se recodifica como:
 - ▶ 0 → Rural
 - ▶ 1 → Urbana
- ▶ $\log(\text{ingreso}) = \log(\text{ingreso})$

Set de datos de la encuesta

```
encuesta_mrp <-  
readRDS("www/05_Mod_Ingreso/encuesta_estan.rds")
```

Table 7: Encuesta estandarizada

dam2	area	logingreso	sexo	anoest	edad	etnia
05360	1	13.27	1	3	3	2
05360	1	13.27	2	2	3	3
05360	1	13.27	1	2	1	2
05360	1	13.27	1	98	1	2
05360	1	13.27	1	98	1	2
05360	1	12.42	1	4	3	2
05360	1	12.42	2	3	3	2
05360	1	12.42	2	2	1	2
05360	1	12.42	1	1	1	2
05360	1	11.99	1	2	4	2

Histograma suavizado del ingreso

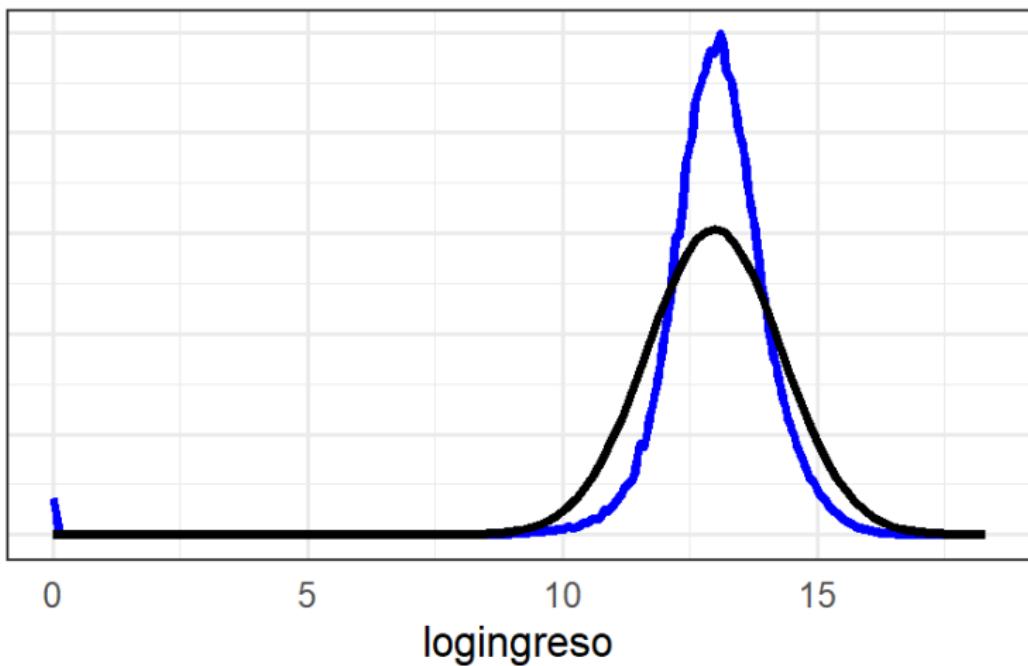


Figure 21: Línea negra distribución normal y Línea azul ingreso suavizado

Creando base con la encuesta agregada

El resultado de agregar la base de dato se muestra a continuación:

```
byAgrega <- c("dam", "dam2", "área", "sexo",
             "anoest", "edad", "etnia")

encuesta_df_agg <-
  encuesta_mrp %>%
  group_by_at(all_of(byAgrega)) %>%
  summarise(n = n(),
            logingreso = mean(logingreso),
            .groups = "drop")
```

Encuesta agregada

El proceso computacional se optimiza al tener la encuesta agregada.

Table 8: Encuesta agregada

dam2	area	sexo	anoest	edad	etnia	n	logingreso
47001	1	2	3	2	3	2636	12.72
11001	1	1	3	2	3	2616	13.25
47001	1	1	3	2	3	2550	12.82
23001	1	2	3	2	3	2530	12.77
11001	1	2	3	2	3	2441	13.16

Ahora, agregamos las covariables

```
encuesta_df_agg <-  
  inner_join(encuesta_df_agg, statelevel_predictors_df)
```

Definiendo el modelo multivariado.

Después de haber ordenado la encuesta, podemos pasar a la definición del modelo.

```
fit <- stan_lmer(  
  logingreso ~      # Log del Ingreso medio (Y)  
    (1 | dam2) +   # Efecto aleatorio (ud)  
    edad +        # Efecto fijo (Variables X)  
    sexo + tasa_desocupacion +  
    luces_nocturnas + cubrimiento_cultivo +  
    cubrimiento_urbano ,  
  weights = n,    # Número de observaciones.  
  data = encuesta_df_agg, # Encuesta agregada  
  verbose = TRUE, # Muestre el avance del proceso  
  chains = 4, # Número de cadenas.  
  iter = 1000) # Número de realizaciones de la cadena  
 saveRDS(fit, file = "Data/fit_ingresos.rds")
```

Validación de la convergencia de las cadenas.

```
library(posterior)
library(bayesplot)
p1 <-
  (mcmc_dens_chains(fit, pars = "sigma") +
    mcmc_áreas(fit, pars = "sigma")) /
  mcmc_trace(fit, pars = "sigma")
ggsave(p1,
       plot = "www/05_Mod_Ingreso/04_Fig_sigma_ing.png" )
```

Cadenas para σ^2

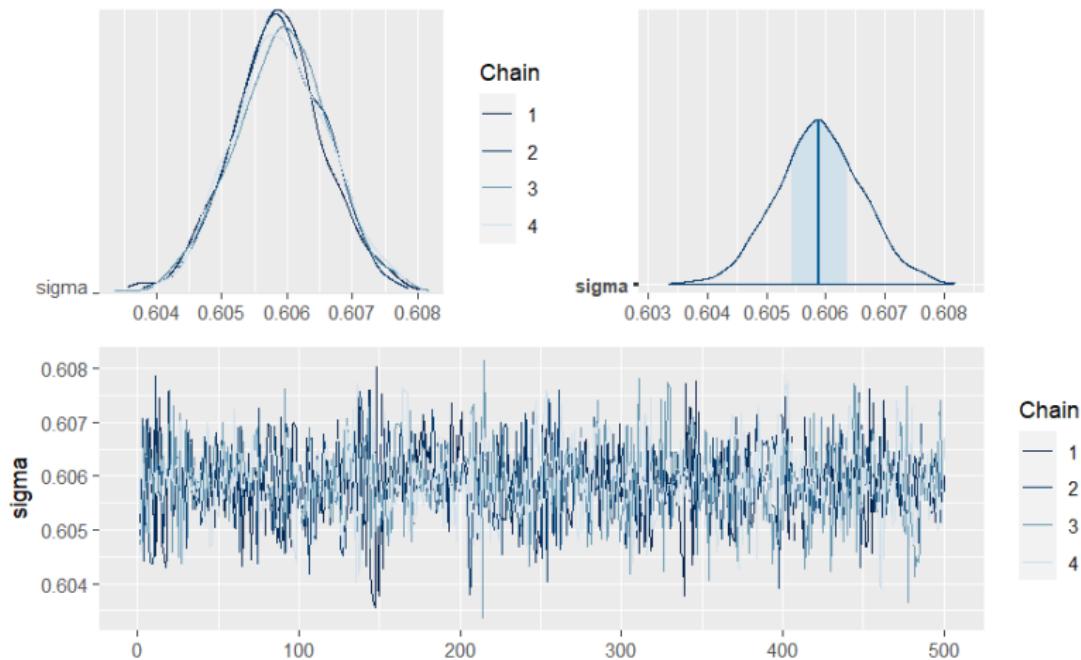


Figure 22: Recorrido de las cadenas

Distribución posterior de los coeficientes

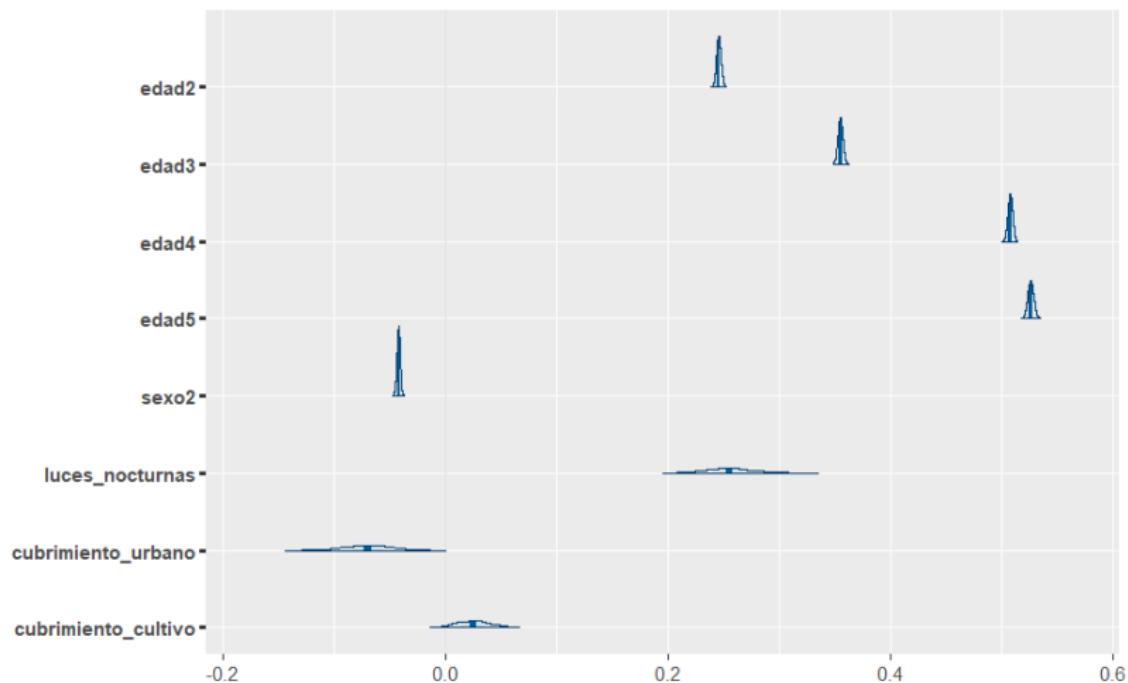


Figure 23: Distribución posterior para los betas

Resultados del modelo en la encuesta

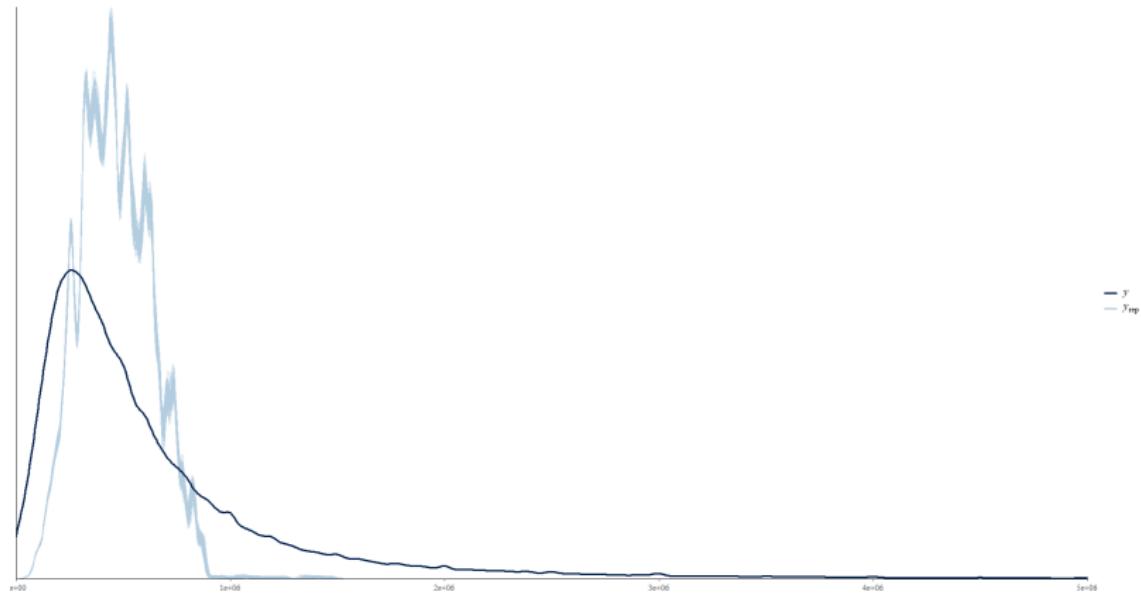


Figure 24: PPC para el ingreso

Resultados del modelo en la encuesta

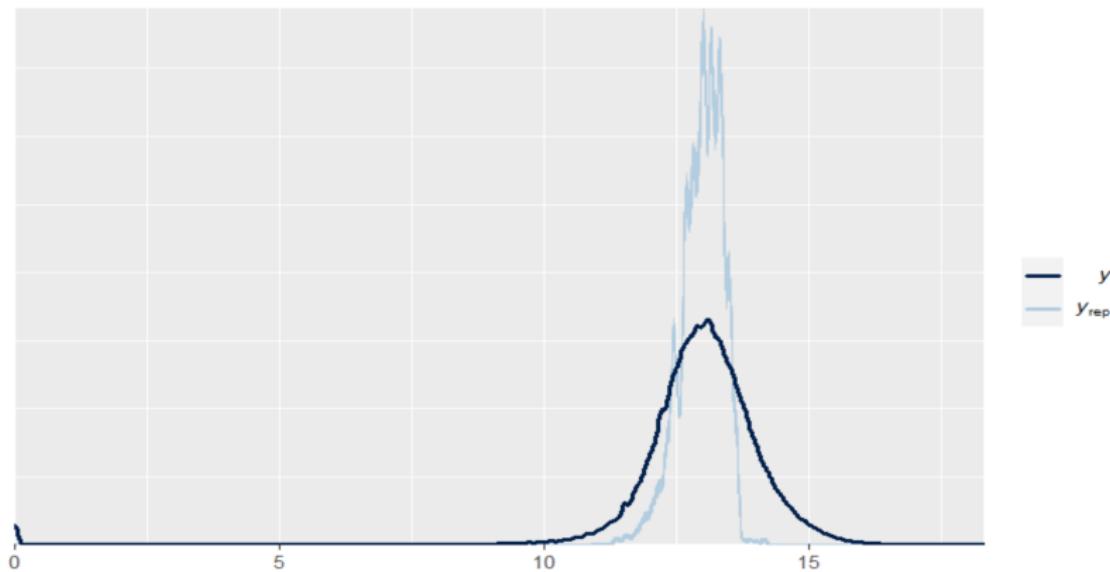


Figure 25: PPC para el log_income

Predictión del ingreso con el modelo de unidad

El proceso de predicción inicia con la lectura del censo agregado que fue estandarizada previamente, luego se une con la base de covariables dando como resultado la siguiente tabla.

```
poststrat_df <-  
  readRDS("www/05_Mod_Ingreso/censo_dam2.rds") %>%  
  left_join(statelevel_predictors_df)
```

Table 9: Censo agregado y covariables

dam2	area	sexo	edad	etnia	anoest	n
05001	0	1	1	1	2	1
05001	0	1	1	1	3	2
05001	0	1	1	1	98	1
05001	0	1	1	2	1	5
05001	0	1	1	2	2	20

Distribución posterior.

Para obtener una distribución posterior de cada observación se hace uso de la función *posterior_epred* de la siguiente forma.

```
epred_mat <- posterior_epred(  
  fit, newdata = poststrat_df,  
  type = "response")
```

Ingreso en términos de líneas de pobreza.

Escribir la estimación del ingreso medio en términos de líneas de pobreza.

```
lp <- encuesta_mrp %>% distinct(área,lp,li)
```

Table 10: Líneas de pobreza

area	lp	li
1	296845	147169
0	200760	127346

```
lp <- inner_join(poststrat_df,lp,by = "área") %>%
  select(lp)
```

```
epred_mat <- (exp(epred_mat)-1)/lp$lp
```

Estimación del ingreso medio nacional

El proceso se reduce a operaciones matriciales, las cuales están organizadas en la función *Aux_Agregado*

```
mrp_estimate_Ingresolp <-
  Aux_Agregado(poststrat = poststrat_df,
                epredmat = epred_mat,
                byMap = NULL)
```

Table 11: Estimación de ingreso medio nacional

Nacional	mrp_estimate	mrp_estimate_se
Nacional	1.931	0.0843

Estimación del ingreso medio dam2

De forma similar es posible obtener los resultados para las divisiones administrativas.

```
mrp_estimate_dam2 <-
  Aux_Agregado(poststrat = poststrat_df,
                epredmat = epred_mat,
                byMap = "dam2")
```

Table 12: Estimación por división administrativas.

dam2	mrp_estimate	mrp_estimate_se
05001	2.604	0.1615
05002	1.028	0.0432
05004	1.066	0.3687
05021	1.089	0.3753
05030	1.575	0.5411

Mapa del ingreso medio con el modelo de unidad

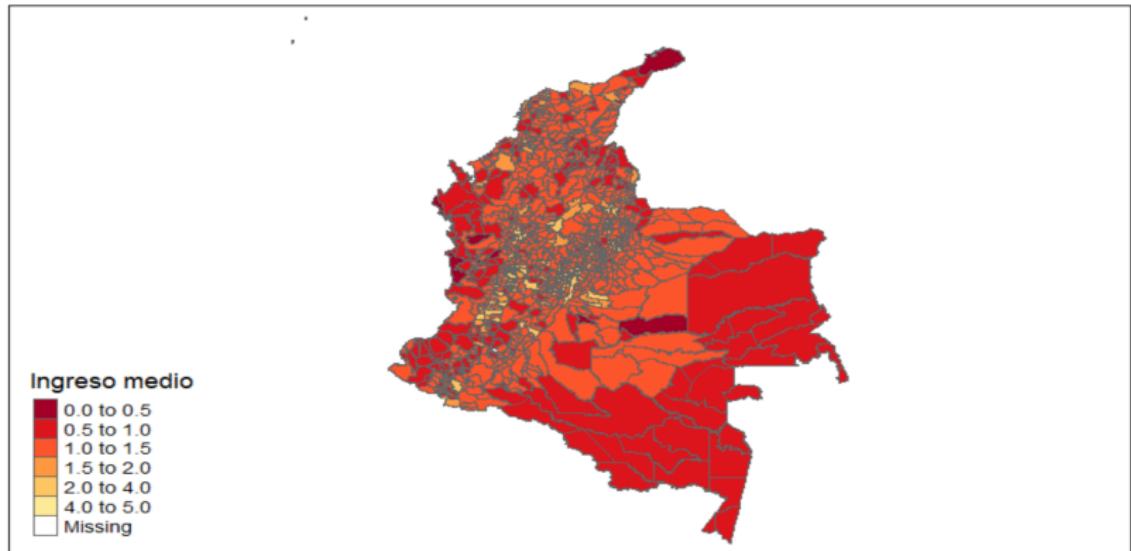


Figure 26: Mapa de ingreso medio con el modelo de unidad

Estimación de la pobreza a partir del ingreso

Sea

$$y_{ji} = \begin{cases} 1 & \text{ingreso}_{ji} \leq lp \\ 0 & \text{e.o.c.} \end{cases}$$

donde ingreso_{ji} representa el ingreso de la i -ésima persona en el j -ésimo post-estrato y lp es un valor límite, en particular la linea de pobreza.

```
epred_mat_pobreza_lp <- (exp(epred_mat)-1) <= lp$lp
```

Estimación de la pobreza

El proceso se simplifica aplicando la función anterior.

```
(mrp_estimate_Ingresolp <-
  Aux_Agregado(poststrat = poststrat_df,
                epredmat = epred_mat_pobreza_lp,
                byMap = NULL)
)
```

Table 13: Estimación de la pobreza

Nacional	mrp_estimate	mrp_estimate_se
Nacional	0.1805	0.0542

Estimación del pobreza por dam2

De forma similar es posible obtener los resultados para las divisiones administrativas.

```
mrp_estimate_dam2 <-
  Aux_Agregado(poststrat = poststrat_df,
                epredmat = epred_mat,
                byMap = "dam2")
```

Table 14: Estimación por división administrativas.

dam2	mrp_estimate	mrp_estimate_se
05001	0.0000	0.0000
05002	0.5176	0.0845
05004	0.5230	0.3123
05021	0.5038	0.3061
05030	0.2005	0.2373

Mapa de pobreza por el modelo de unidad

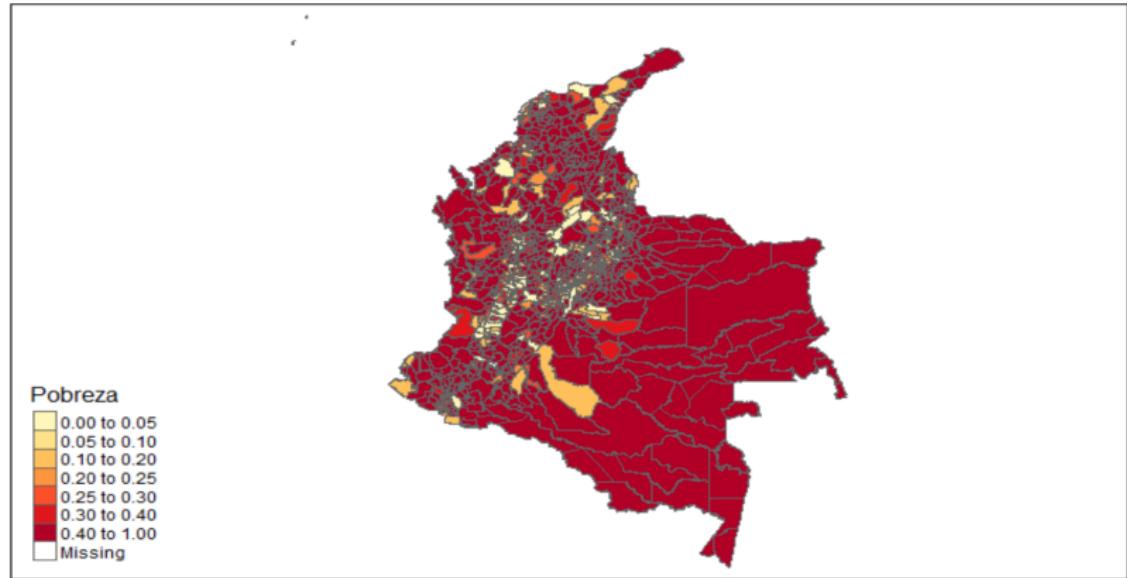


Figure 27: Mapa de la pobreza a partir ingreso medio

Modelo de unidad para la estimación de la pobreza

- ▶ La regresión logística se usa cuando la variable dependiente es binaria, ya que permite estimar la probabilidad del evento estudiado.
- ▶ Para obtener estimaciones de probabilidad, se realiza una transformación logarítmica conocida como *logit*.
- ▶ El logit se calcula como el logaritmo de la probabilidad de éxito dividido por la probabilidad de fracaso:

$$\ln \left(\frac{\theta}{1 - \theta} \right)$$

donde θ es la probabilidad de éxito.

Modelo de unidad con respuesta binaria

- ▶ Se emplea un modelo de regresión logística de efectos aleatorios para relacionar la expectativa θ_{ji} de esta variable con covariables disponibles x_{ji} y el efecto aleatorio u_d .
- ▶ El modelo se expresa como: $\ln\left(\frac{\theta_{ji}}{1-\theta_{ji}}\right) = x_{ji}^T \beta + u_d$.
- ▶ Los coeficientes β son los efectos fijos de las variables sobre las probabilidades, y u_d son efectos aleatorios.

Distribuciones Previas

Las distribuciones previas son no informativas y se asumen como:

- $\beta_k \sim N(0, 1000)$.
- $\sigma_u^2 \sim IG(0.0001, 0.0001)$.

Proceso de estimación

- ▶ Estimar la proporción de personas por debajo de la línea de pobreza: $P_d = \frac{\sum_{U_d} y_{di}}{N_d}$.
- ▶ El estimador se calcula como: $\hat{P} = \frac{\sum_{s_d} y_{di} + \sum_{s_d^c} \hat{y}_{di}}{N_d}$.
- ▶ Donde \hat{y}_{di} es el valor esperado de y_{di} bajo el modelo.

Estimación en R

El proceso inicia con la definición de la pobreza haciendo uso de la línea de pobreza definida en CEPAL así

```
encuesta_mrp %>% mutate(  
  pobreza = ifelse(ingreso < lp, 1, 0))
```

Creando base con la encuesta agregada

En forma similar al modelo del ingreso, ahora se realiza un conteo de las personas que están por debajo de la linea de pobreza agregando por algunas variables.

```
encuesta_df_agg <-  
  encuesta_mrp %>%      # Encuesta  
  group_by_at(all_of(byAgrega)) %>%  
  summarise(n = n(), # Número de observaciones  
            # conteo de personas con características similares.  
            pobreza = sum(pobreza),  
            no_pobreza = n-pobreza,  
            .groups = "drop") %>%  
  arrange(desc(pobreza))          # Ordenar la ba
```

Tabla agregada

El resultado de agregar la base de dato se muestra a continuación:

Table 15: Conteo de personas en condición de pobreza

dam2	area	sexo	anoest	edad	etnia	pobreza	no_pobreza
47001	1	2	3	2	3	1048	1588
27001	1	2	3	2	2	993	831
20001	1	2	3	2	3	953	1258
23001	1	2	3	2	3	909	1621
47001	1	1	3	2	3	870	1680

Ahora incorporan las covariables.

```
encuesta_df_agg %<>>%  
  inner_join(statelevel_predictors_df)
```

Modelo de unidad en STAN

Después de haber ordenado la encuesta, podemos pasar a la definición del modelo.

```
fit <- stan_glmer(  
  cbind(pobreza, no_pobreza) ~  
    (1 | dam2) +      # Efecto aleatorio (ud)  
    edad +           # Efecto fijo (Variables X)  
    sexo + tasa_desocupacion +  
    luces_nocturnas + cubrimiento_cultivo +  
    cubrimiento_urbano ,  
    data = encuesta_df_agg, # Encuesta agregada  
    verbose = TRUE,       # Muestre el avance del proceso  
    chains = 4,           # Número de cadenas.  
    iter = 100, cores = 4,  
    family = binomial(link = "logit")  
  )  
saveRDS(fit, file = "www/06_Mod_Pobreza/fit_pobreza.rds")
```

Distribución posterior de los coeficientes

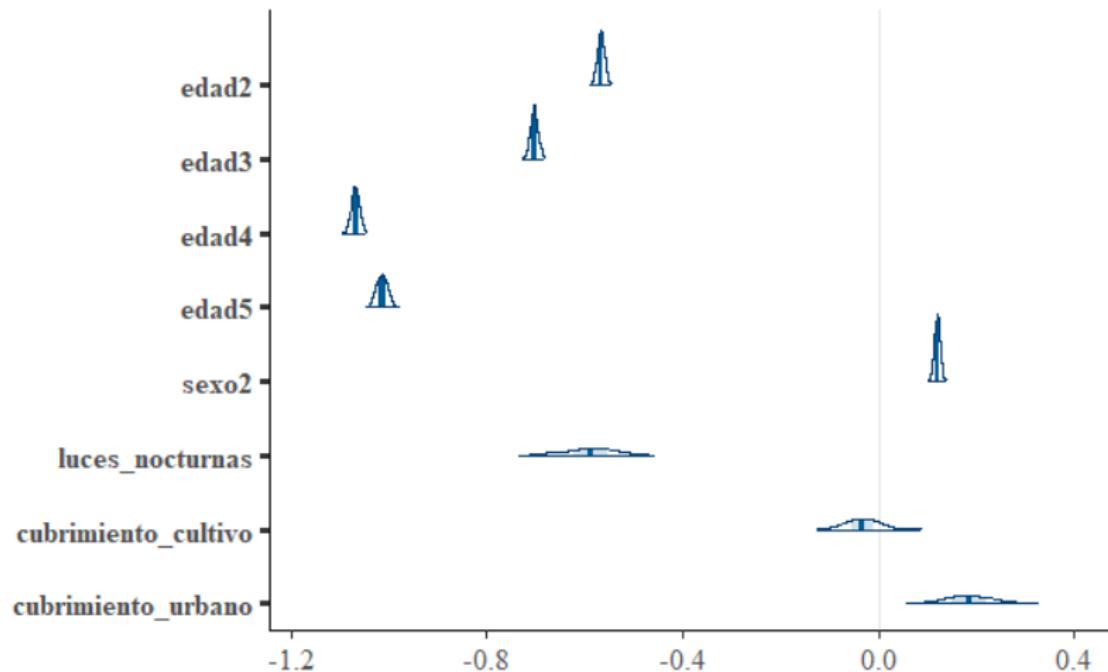


Figure 28: Distribución posterior de los coeficientes

Seguimiento de las cadenas para los coeficientes

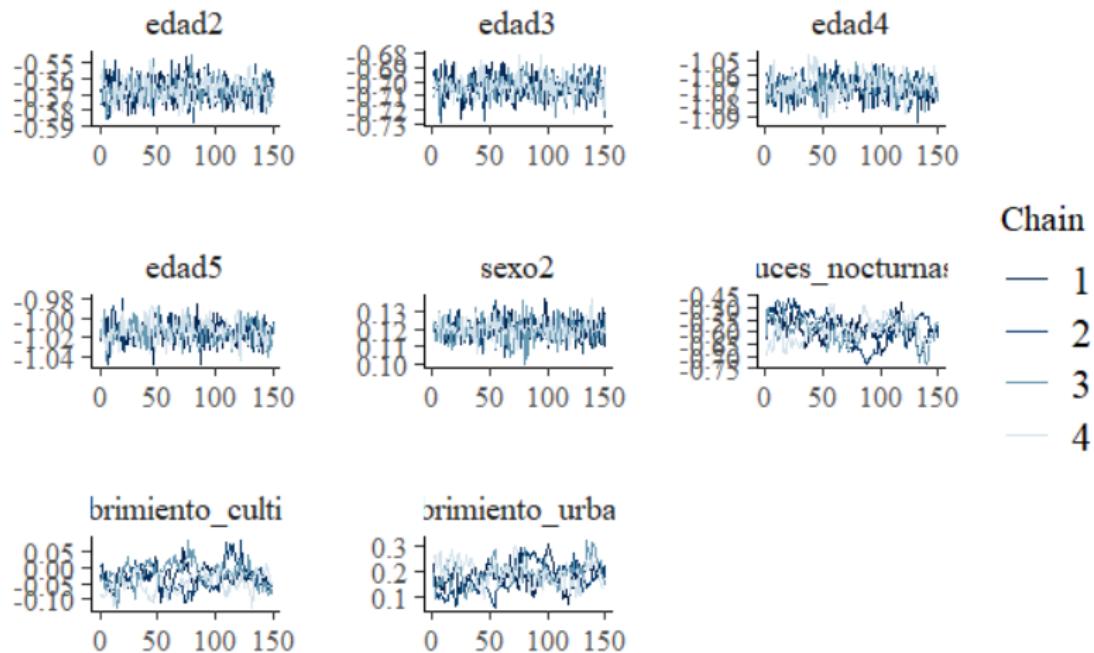


Figure 29: Cadenas de los coeficientes

Resultados del modelo en la encuesta

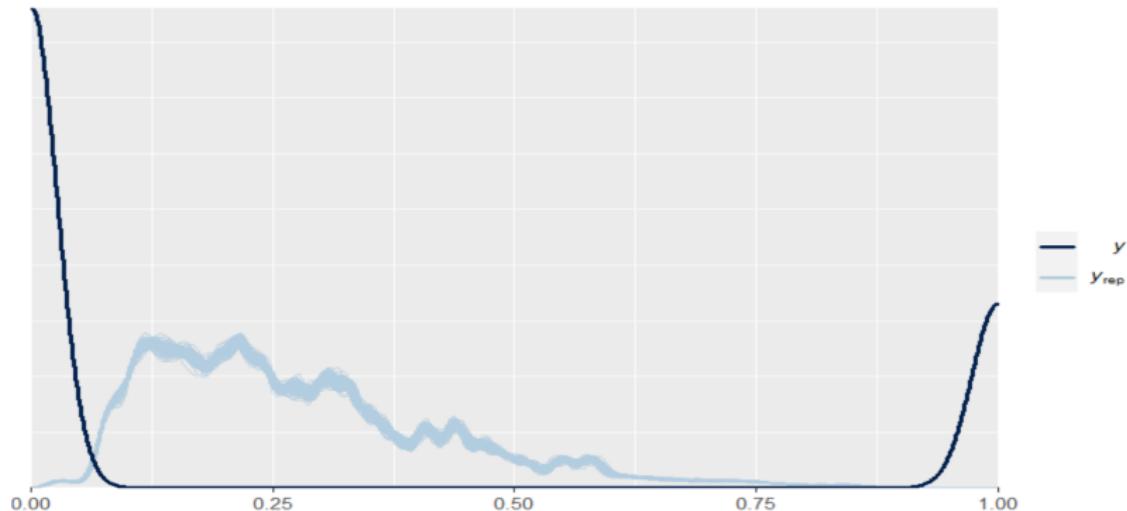


Figure 30: PPC para el ingreso

Predicción del pobreza con el modelo de unidad

El proceso de predicción inicia con la lectura del censo agregado que fue estandarizada previamente, luego se une con la base de covariables dando como resultado la siguiente tabla.

```
poststrat_df <-  
  readRDS("www/06_Mod_Pobreza/censo_dam2.rds") %>%  
  left_join(statelevel_predictors_df)
```

Table 16: Censo agregado y covariables

dam2	area	sexo	edad	etnia	anoest	n
05001	0	1	1	1	2	1
05001	0	1	1	1	3	2
05001	0	1	1	1	98	1
05001	0	1	1	2	1	5
05001	0	1	1	2	2	20

Distribución posterior.

Para obtener una distribución posterior de cada observación se hace uso de la función *posterior_epred* de la siguiente forma.

```
epred_mat <- posterior_epred(  
  fit, newdata = poststrat_df,  
  type = "response")
```

Estimación de la tasa de pobreza

De forma similar al modelo de ingreso se hace uso de la función *Aux_Agregado* para tener las estimaciones de tasa de pobreza.

```
(mrp_estimate_Ingresolp <-  
  Aux_Agregado(poststrat = poststrat_df,  
                epredmat = epred_mat,  
                byMap = NULL)  
) %>% tba()
```

Table 17: Estimación de la tasa de pobreza

Nacional	mrp_estimate	mrp_estimate_se
Nacional	0.2849	0.0286

Estimación de la tasa de pobreza por dam2

De forma similar es posible obtener los resultados para las divisiones administrativas del país.

```
mrp_estimate_dam2 <-
  Aux_Agregado(poststrat = poststrat_df,
                epredmat = epred_mat,
                byMap = "dam2")
```

dam2	mrp_estimate	mrp_estimate_se
05001	0.1532	0.0021
05002	0.4189	0.0300
05004	0.4747	0.1545
05021	0.4638	0.1539
05030	0.2981	0.1337

Mapa de pobreza estimado con el modelo de unidad

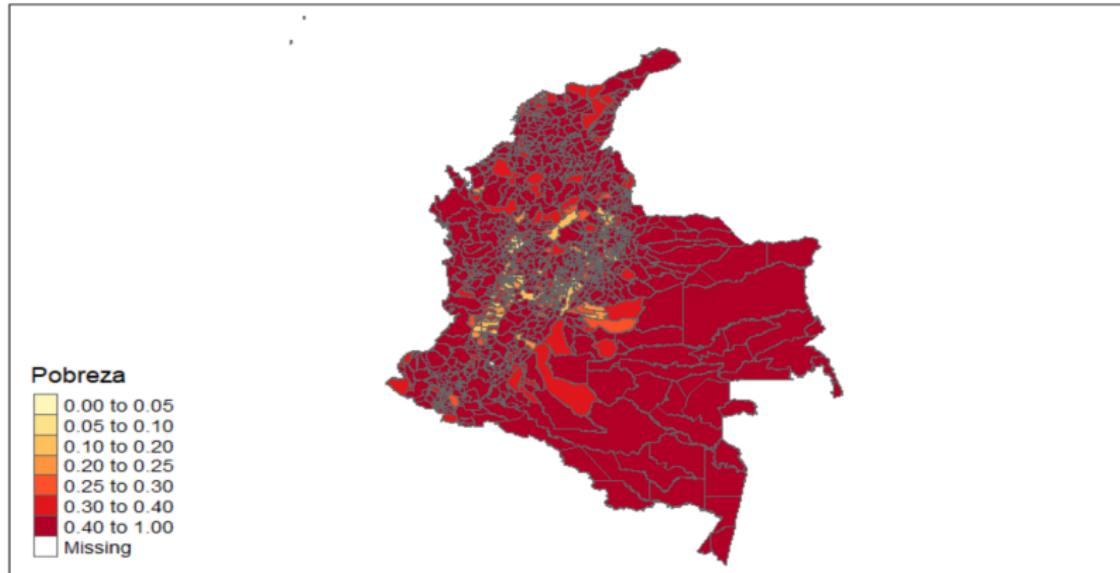


Figure 31: Mapa de la pobreza modelo de unidad

Modelo de unidad Índice de Privación Multidimensional (IPM)

Introducción

- ▶ La pobreza es un tema crucial en la agenda nacional e internacional, como lo demuestra el primer objetivo de la agenda 2030 para el Desarrollo Sostenible.
- ▶ Tradicionalmente, se mide la pobreza de manera unidimensional, basada en ingresos y gastos.
- ▶ Abordar la pobreza desde una perspectiva multidimensional permite capturar una gama más amplia de factores que afectan la calidad de vida.

Índice de Pobreza Multidimensional (IPM)

- ▶ El IPM es una medida que evalúa la pobreza considerando múltiples dimensiones de bienestar.
- ▶ Se calcula mediante ponderaciones y umbrales en función de diferentes indicadores de calidad de vida.
- ▶ El IPM es una variante de la metodología FGT (Foster, Greer y Thorbecke, 1984) utilizada para medir la pobreza unidimensional.

Ecuación del IPM

- Se expresa como un promedio de puntuación de privación censurada, como se detalla en las siguientes ecuaciones:

$$IPM = \frac{1}{N} \sum_{i=1}^N c_i(z)$$

Donde: - N es el número de individuos u hogares en la población.
- $c_i(z)$ es el puntaje de privación censurado de la observación i .

Calculo de $c_i(z)$

La forma de obtener $c_i(z)$ esta dado por la siguiente ecuación:

- ▶ Si $q_i \geq z$ entonces c_i será igual a q_i
- ▶ Si $q_i < z$ entonces c_i será igual a 0

Con: - $q_i = \sum_{k=1}^K w_k \cdot y_i^k$, donde K es el número de dimensiones o indicadores de la privación, w_k es el ponderador asociado a la dimensión k , y y_i^k es una variable binaria.

Componentes del IPM:

1. Headcount Ratio (H):

- ▶ Mide la proporción de personas privadas en al menos una dimensión de pobreza.
- ▶ Se calcula como el número de personas privadas en al menos una dimensión sobre la población total.
- ▶
$$H = \frac{1}{N} \sum_{i=1}^N I(q_i \geq z) = \frac{N(z)}{N}$$
 donde
$$N(z) = \sum_{i=1}^N I(q_i \geq z)$$

2. Intensity of Deprivation (A):

- ▶ Mide la intensidad promedio de privación entre las personas privadas.
- ▶ Se calcula como el promedio de los indicadores de privación para aquellas personas que están privadas en al menos una dimensión.
- ▶
$$A = \sum_{i=1}^N \frac{c_i(z)}{N(z)}$$

Cálculo del IPM apartir de H y A

- ▶ El IPM se obtiene multiplicando los valores de H y A.
- ▶ Matemáticamente, se expresa como el promedio de puntuación de privación censurada.

$$IPM = \frac{N(z)}{N} \times \sum_{i=1}^N \frac{c_i(z)}{N(z)} = \frac{1}{N} \sum_{i=1}^N c_i(z)$$

Modelo de unidad para el IPM

- ▶ En muchas aplicaciones, la variable de interés en áreas pequeñas es binaria, es decir, y_{dj} toma valores de 0 o 1, representando la ausencia o presencia de una característica específica.
- ▶ El objetivo de estimación en cada dominio $d = 1, \dots, D$ es la proporción $\theta_d = \frac{1}{N_d} \sum_{i=1}^{N_d} y_{di}$ de la población que presenta esta característica.
- ▶ El logit de θ_{di} se define como

$$\ln \left(\frac{\theta_{di}}{1 - \theta_{di}} \right) = \eta_{di} = x_{di}^T \beta + u_d$$

donde β es un vector de parámetros de efecto fijo y u_d es un efecto aleatorio específico del área para el dominio d con $u_d \sim N(0, \sigma_u^2)$.

Modelo de unidad para el IPM

- ▶ Los u_d son independientes, y $y_{di} \mid u_d \sim Bernoulli(\theta_{di})$ con $E(y_{di} \mid u_d) = \theta_{di}$ y $Var(y_{di} \mid u_d) = \sigma_{di}^2 = \theta_{di}(1 - \theta_{di})$.
- ▶ x_{di}^T representa un vector de $p \times 1$ de valores de p variables auxiliares.
- ▶ Entonces, θ_{di} se puede expresar como:

$$\theta_{di} = \frac{\exp(x_{di}^T \beta + u_d)}{1 + \exp(x_{di}^T \beta + u_d)}$$

El modelo se estima para cada dimensión.

Distribuciones previas

Como es tradicional se usan distribuciones previas no informativas

- ▶ $\beta_k \sim N(0, 10000)$
- ▶ $\sigma_u^2 \sim IG(0.0001, 0.0001)$

Estimación del IPM usando los modelos de unidad

- ▶ Estimar la proporción de personas que presentan la k -ésima carencia, es decir, $P_d = \frac{\sum_{U_d} c_{di}(z)}{N_d}$.
- ▶ El estimador de P se calcula como:

$$\hat{P}_d = \frac{\sum_{s_d} c_{di}(z) + \sum_{s_d^c} \hat{c}_{di}(z)}{N_d}$$

donde $\hat{c}_{di}(z)$ se define como:

- ▶ Si $\hat{q}_{di} \geq z$ entonces c_{di} será igual a \hat{q}_{di}
- ▶ Si $\hat{q}_{di} < z$ entonces c_{di} será igual a 0

Estimación de q_{di}

La estimación de \hat{q} esta dada por

$$\hat{q}_{di} = \sum_{k=1}^K w_k \cdot \hat{y}_{di}^k$$

donde

$$\hat{y}_{di}^k = E_{\mathcal{M}}(y_{di}^k | x_d, \beta)$$

- Así, se obtiene el estimador de P para cada dominio d .

Estimación de θ_{di}^k

La estimación de θ_{di}^k refleja la probabilidad de que una unidad específica i en el dominio d obtenga el valor 1 en la dimensión k . Para llevar a cabo esta estimación, seguimos el siguiente procedimiento:

$$\bar{Y}_d^k = \theta_d^k = \frac{1}{N_d} \sum_{i=1}^{N_d} y_{di}^k$$

Aquí, y_{di}^k puede tomar los valores 0 ó 1, representando la ausencia (o no) de una característica específica.

Estimación de θ_{di}^k

Dividir la suma en dos partes: s_d , que representa las unidades observadas en una muestra, y s_d^c , que son las unidades no observados. Por lo tanto,

$$\bar{Y}_d^k = \theta_d^k = \frac{1}{N_d} \left(\sum_{s_d} y_{di}^k + \sum_{s_d^c} y_{di}^k \right)$$

Estimación de θ_{di}^k

Mediante un modelo de unidad es posible realizar la predicción de y_{di}^k para las unidades no observadas. De esta manera, el estimador de θ_d^k se expresa como:

$$\hat{\theta}_d^k = \frac{1}{N_d} \left(\sum_{s_d} y_{di}^k + \sum_{s_d^c} \hat{y}_{di}^k \right)$$

Donde,

$$\hat{y}_{di}^k = E_{\mathcal{M}} (y_{di}^k | x_d, \beta)$$

Estimación de θ_{di}^k

La estimación $\hat{\theta}_d^k$ se simplifica a:

$$\hat{\theta}_d^k = \frac{1}{N_d} \sum_{i=1}^{N_d} \hat{y}_{di}^k$$

Este enfoque permite estimar la probabilidad θ_d^k en el dominio d en la dimensión k utilizando predicciones y datos disponibles en lugar de contar con información individual detallada para todos los casos.

Predicción de los “Hard Estimates

- ▶ Hobza y Morales (2016) definen los “hard estimates” como valores binarios (0 o 1) que indican de manera precisa si un individuo tiene o no una característica específica en relación con cada indicador de privación multidimensional.
- ▶ La estimación de θ_{di}^k refleja la probabilidad de que una unidad específica i en el dominio d obtenga el valor 1 en la dimensión k .
- ▶ Por lo tanto, se define $\hat{y}_{di}^k \sim Bernoulli(\hat{\pi}_{di}^k)$, donde \hat{y}_{di}^k son las estimaciones “hard”.

Estimación Puntual del IPM

El procedimiento propuesto para estimar el IPM es el siguiente:

1. Utilice los datos de la muestra para ajustar un modelo logit Bernoulli a nivel de unidad para cada indicador. Esto se logra mediante el uso del algoritmo de Markov Chain Monte Carlo (MCMC) con L iteraciones.
2. Para cada dimensión k a la cual se estimo un modelo logit Bernoulli a nivel de unidad con L iteraciones, realice la predicción de los valores \hat{y}_{di}^k para cada individuo en el censo. Esto generará L realizaciones aleatorias de \hat{y}_{di}^k .
3. Denotemos como \hat{y}_{di}^{kl} a la l -ésima realización aleatoria de la dimensión k para el individuo i en el dominio d . Calculamos $q_{di}^l = \sum_{k=1}^K w_k \cdot \hat{y}_{di}^{kl}$.

Estimación Puntual del IPM

Apartir de los valores calculados para q_{di} se puede calcular H_d^l , A_d^l y IPM_d^l utilizando las ecuaciones:

$$IPM_d^l = \frac{1}{N_d} \sum_{i=1}^{N_d} c_{di}^l(z)$$

$$H_d^l = \frac{1}{N_d} \sum_{i=1}^{N_d} I(q_{di}^l \geq z) = \frac{N_d^l(z)}{N_d}$$

y

$$A_d^l = \sum_{i=1}^{N_d} \frac{c_{di}^l(z)}{N_d^l(z)}$$

Estimación Puntual del IPM

4. La estimación puntual de H_d , A_d y IPM_d en cada área pequeña d se calcula tomando el promedio sobre las L iteraciones:

$$\hat{H}_d = \frac{1}{L} \sum_{l=1}^L H_d^l,$$

$$\hat{A}_d = \frac{1}{L} \sum_{l=1}^L A_d^l$$

y

$$\widehat{IPM}_d = \frac{1}{L} \sum_{l=1}^L IPM_d^l$$

Estimación de la varianza para el IPM

5. Dada que el modelo se estimó usando el algoritmo MCMC, es posible tener la estimación del error de estimación, de esta forma:

$$\widehat{Var}(\hat{H}_d) = \frac{1}{L} \sum_{l=1}^L (H_d^l - \hat{H}_d)^2,$$

$$\widehat{Var}(\hat{A}_d) = \frac{1}{L} \sum_{l=1}^L (A_d^l - \hat{A}_d)^2$$

y

$$\widehat{Var}(\widehat{IPM}_d) = \frac{1}{L} \sum_{l=1}^L (IPM_d^l - \widehat{IPM}_d)^2$$

Índice de Privación Multidimensional en Colombia.

En Colombia se tiene $K = 9$ indicadores que se miden como privaciones: $y_{di}^k = 1$ si la persona tiene la privación y $y_{di}^k = 0$ si la persona no ha tenido la privación.

El índice requiere información para cada individuo $i = 1, \dots, N_d$ en los dominios $d = 1, \dots, D$, donde N_d denota el tamaño de la población del dominio d .

Para este estudio, utilizamos el valor de 0.4 para z , es decir, $I(\cdot)$ es igual a 1 cuando $q_{di} \geq 0.4$. El valor de q_{di} el dominio d se calcula como:

$$q_{di} = \frac{1}{16}(y_{di}^1 + y_{di}^2 + y_{di}^3 + y_{di}^4) + \frac{1}{12}(y_{di}^5 + y_{di}^6 + y_{di}^7) + \frac{1}{4}(y_{di}^8 + y_{di}^9)$$

Privaciones calculadas para Colombia.

- a. y_{di}^1 = Privación en material de construcción de la vivienda
- b. y_{di}^2 = Hacinamiento en el hogar.
- c. y_{di}^3 = Acceso al servicio de Internet.
- d. y_{di}^4 = Acceso al servicio energía eléctrica.
- e. y_{di}^5 = Privación en saneamiento.
- f. y_{di}^6 = Privación de acceso al agua potable.
- g. y_{di}^7 = Privación en salud.
- h. y_{di}^8 = Privación de la educación.
- i. y_{di}^9 = Privación del empleo y la protección social.

Dimensiones de las privaciones.

Las privaciones anteriores se agrupan por dimensiones así:

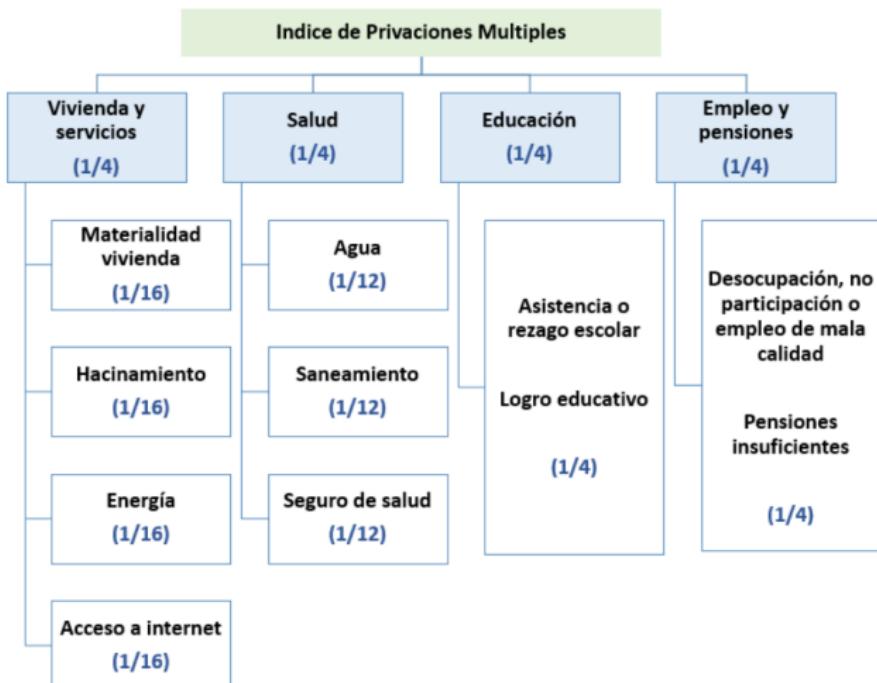


Figure 32: Dimensiones y pesos del IPM

Encuesta de hogares con indicadores de las privaciones.

En la siguiente tabla observan una muestra de las y_{di}^k para Colombia.

Table 18: Índices de provaciones en Colombia

dam2	nbi_matviv_ee	nbi_hacina_ee	nbi_tic	nbi_agua_ee
05360	0	0	0	0
05360	0	0	0	0
05360	0	0	0	0
05360	0	0	0	0
05360	0	0	0	0
05360	0	0	1	0
05360	0	0	1	0
05360	0	0	1	0
05360	0	0	1	0
05360	0	0	1	0

Proceso para agrupar la encuestas.

Lectura de la encuesta y definición de variables para agrupar

Proceso para agrupar la encuestas.

El proceso se repite para cada una de las privaciones, por tanto se automatiza de la siguiente forma:

```
encuesta_df <- map(
  setNames(names_ipm, names_ipm),
  function(y) {
    encuesta_ipm$temp <- as.numeric(encuesta_ipm[[y]])
    encuesta_ipm %>%
      group_by_at(all_of(byAgrega)) %>%
      summarise( n = n(), yno = sum(temp),
                 ysi = n - yno, .groups = "drop"
      ) %>%
      inner_join(statelevel_predictors_df,
                 by = c("dam", "dam2"))
  })
  saveRDS(encuesta_df,
           "www/07_Mod_IPM/03_tabla_encuesta_agg.rds")
```

Muestra de las bases

La base resultante quedan de la siguiente forma:

Table 19: Privaci'on en material de construcci'on de la vivienda

dam2	area	sexo	etnia	anoest	n	yno	ysi	area1	sexo2
05001	0	1	3	1	3	0	3	0.9832	0.5299
05001	0	1	3	1	1	0	1	0.9832	0.5299
05001	0	1	3	1	1	1	0	0.9832	0.5299
05001	0	1	3	2	3	0	3	0.9832	0.5299
05001	0	1	3	2	2	0	2	0.9832	0.5299
05001	0	1	3	2	1	0	1	0.9832	0.5299
05001	0	1	3	2	3	0	3	0.9832	0.5299
05001	0	1	3	2	2	0	2	0.9832	0.5299
05001	0	1	3	3	6	1	5	0.9832	0.5299
05001	0	1	3	3	1	0	1	0.9832	0.5299

Definir el modelo

```
names_cov <- statelevel_predictors_df %>%
  dplyr::select(-dam,-dam2) %>% names()
names_cov <- c("sexo","área",names_cov[16:19])
efec_aleat <-
  paste0("(1|",  c("dam", "etnia"), ")",
         collapse = "+")
formula_mod <- formula(paste(
  " cbind(yno, ysi) ~", efec_aleat,
  "+",  paste0(names_cov,
               collapse = " + "))
))

formula_mod
```

```
cbind(yno, ysi) ~ (1 | dam) + (1 | etnia) +
  sexo + área + tasa_desocupacion +
  luces_nocturnas + cubrimiento_cultivo +
  cubrimiento_urba
```

Ejecutando los modelos

```
plan(multisession, workers = 4)

fit <- future_map(encuesta_df, function(xdat){
  stan_glmer(formula = formula_mod ,
  family = binomial(link = "logit"),
  data = xdat,
  cores = 4,
  chains = 4,
  iter = 500
),
.progress = TRUE)

saveRDS(object = fit, "Data/fits_IPM.rds")
```

Proceso para la predicción θ_{di}^{kl}

- ▶ Los modelos fueron compilados de manera separada, por tanto, disponemos de un objeto .rds por cada privación que componen el IPM.
- ▶ El proceso se ilustra el proceso para la privación en agua, pero es igual para las restantes privaciones.
- ▶ La predicción de los modelos se hace sobre la base del censo.

Predicción en el censo

```
censo_ipm <- readRDS("www/07_Mod_IPM/04_tabla_censo.rds")
fit_agua <-
  readRDS(file = "www/07_Mod_IPM/Modelos/fit_agua.rds")

epred_mat_agua <- posterior_epred(
  fit_agua,
  newdata = poststrat_df,
  type = "response",
  allow.new.levels = TRUE
)
```

Definiendo los hard estimates

```
epred_mat_agua_dummy <-
  rbinom(
    n = nrow(epred_mat_agua) * ncol(epred_mat_agua) , 1,
    epred_mat_agua)

epred_mat_agua_dummy <- matrix(
  epred_mat_agua_dummy,
  nrow = nrow(epred_mat_agua),
  ncol = ncol(epred_mat_agua)
)
saveRDS(epred_mat_agua_dummy,
        "www/07_Mod_IPM/Dummys/epred_mat_agua_dummy.rds")
```

Calculando q_{di}^l

El calculo de q_{id}^l es una simple operación matricial.

```
chain_q <-
  # Vivienda y servicios
  (1 / 16) * (
    epred_mat_material_dummy +
    epred_mat_hacinamiento_dummy +
    epred_mat_energia_dummy +
    epred_mat_tic_dummy
  ) +
  # Salud
  (1 / 12) * (epred_mat_agua_dummy +
    epred_mat_saneamiento_dummy +
    epred_mat_salud_dummy) +
  # Educación
  (1 / 4) * epred_mat_educacion_dummy +
  # Empleo
  (1 / 4) * epred_mat_empleo_dummy
```

Calculando $I(q_{di}^l \geq z)$ y $c_{di}^l(z)$

Ahora, es posible tener el calculo de $I(q_{di}^l \geq z)$, tomando como umbral $z = 0.4$.

```
chain_Ind <- chain_q  
chain_Ind[chain_Ind < 0.4] <- 0  
chain_Ind[chain_Ind != 0] <- 1
```

seguidamente calculamos $c_{di}^l(z)$

```
chain_ci <- matrix(0,nrow = nrow(chain_q),  
                     ncol = ncol(chain_q))  
chain_ci[chain_Ind == 1] <- chain_q[chain_Ind == 1]
```

Resltados obtenidos en las primeras iteraciones

Table 20: Cadenas obtenidas

q1	q2	Ind1	Ind2	c1	c2	N
0.1875	0.6458	0	1	0.0000	0.6458	1
0.7083	0.3958	1	0	0.7083	0.0000	1
0.7708	0.5208	1	1	0.7708	0.5208	9
0.6250	0.4583	1	1	0.6250	0.4583	1
0.5625	0.3750	1	0	0.5625	0.0000	5
0.3125	0.6250	0	1	0.0000	0.6250	22
0.2500	0.3750	0	0	0.0000	0.0000	9
0.6250	0.2083	1	0	0.6250	0.0000	76
0.3958	0.7083	0	1	0.0000	0.7083	796
0.7083	0.6250	1	1	0.7083	0.6250	3549

Estimaciones desagregadas del IPM

Para realizar las estimaciones desagregadas se desarrolló una función que facilita el cálculo, por ejemplo, división administrativa (*dam2*)

```
source("www/07_Mod_IPM/06_Estimar_ipm.R")
ipm_dam2 <- estime_IPM(
  poststrat = censo_ipm,
  chain_ci = chain_ci,
  chain_ind = chain_ind,
  byMap = "dam2"
) %>% data.frame()
```

Estimaciones desagregadas del IPM

Table 21: Estimaciones por división administrativa

dam2	H	H_sd	A	A_sd	IPM	IPM_sd
05001	0.1247	0.0746	0.5502	0.0295	0.0684	0.0411
05002	0.6014	0.0781	0.6229	0.0258	0.3746	0.0510
05004	0.6555	0.0767	0.6291	0.0260	0.4124	0.0511
05021	0.5196	0.0807	0.6152	0.0236	0.3194	0.0494
05030	0.4771	0.0771	0.6029	0.0225	0.2874	0.0463
05031	0.5242	0.0780	0.6174	0.0231	0.3233	0.0471
05034	0.5397	0.0742	0.6161	0.0226	0.3323	0.0460
05036	0.5246	0.0785	0.6092	0.0242	0.3196	0.0496
05038	0.6856	0.0878	0.6300	0.0269	0.4320	0.0587
05040	0.5796	0.0799	0.6228	0.0231	0.3609	0.0511

Estimaciones por privación del IPM

- ▶ Es fundamental analizar cada dimensión individualmente.
- ▶ Esto permite comprender la complejidad de la pobreza y diseñar estrategias efectivas.
- ▶ Se utilizan “hard estimates” para calcular las estimaciones en cada privación.
- ▶ El proceso se aplica de manera similar a todas las privaciones

Proceso de estimación

Para agilizar el proceso de calculo se define crea la función **agregado_dim_ipm** que hace los cálculos. La forma de uso es la siguiente.

```
source("www/07_Mod_IPM/07_Fun_agregado.r")  
  
epred_mat_agua_dummy <-  
readRDS("www/07_Mod_IPM/Dummys/epred_mat_agua_dummy.rds")  
  
datos_dam_agua <-  
agregado_dim_ipm(poststrat = censo_ipm,  
                  epredmat = epred_mat_agua_dummy,  
                  byMap = "dam2")
```

Estimación por división administrativa

Table 22: Estimación por dam2 de la privación de agua

dam2	estimate	estimate_se
05001	0.0012	0.0082
05002	0.1645	0.0753
05004	0.2000	0.0772
05021	0.1319	0.0572
05030	0.0925	0.0491
05031	0.1352	0.0562
05034	0.1391	0.0570
05036	0.1227	0.0625
05038	0.2084	0.0936
05040	0.1636	0.0652

Resultado para todas las privaciones.

Table 23: Estimacion puntual por municipio y dimension

dam2	Agua	Educacion	Empleo	Energia	Internet
05001	0.0012	0.2518	0.4725	0.0025	0.4783
05002	0.1645	0.6267	0.7798	0.2160	0.8866
05004	0.2000	0.6593	0.7985	0.2673	0.9089
05021	0.1319	0.5715	0.7442	0.1656	0.8532
05030	0.0925	0.5443	0.7119	0.1046	0.8295
05031	0.1352	0.5731	0.7481	0.1722	0.8542
05034	0.1391	0.5837	0.7512	0.1703	0.8599
05036	0.1227	0.5818	0.7383	0.1272	0.8660
05038	0.2084	0.6816	0.8165	0.2823	0.9242
05040	0.1636	0.6089	0.7721	0.2093	0.8784

Resultado para todas las privaciones.

Table 24: Error de estimacion por municipio y dimension

dam2	Agua_se	Educacion_se	Empleo_se	Energia_se	Internet_se
05001	0.0082	0.0981	0.1089	0.0081	0.1138
05002	0.0753	0.0820	0.0718	0.0775	0.0470
05004	0.0772	0.0883	0.0697	0.0857	0.0428
05021	0.0572	0.0877	0.0777	0.0608	0.0622
05030	0.0491	0.0798	0.0777	0.0529	0.0604
05031	0.0562	0.0827	0.0764	0.0564	0.0614
05034	0.0570	0.0796	0.0710	0.0609	0.0549
05036	0.0625	0.0825	0.0719	0.0616	0.0508
05038	0.0936	0.0931	0.0797	0.0969	0.0397
05040	0.0652	0.0836	0.0750	0.0686	0.0531

Mapa de los componentes del IPM

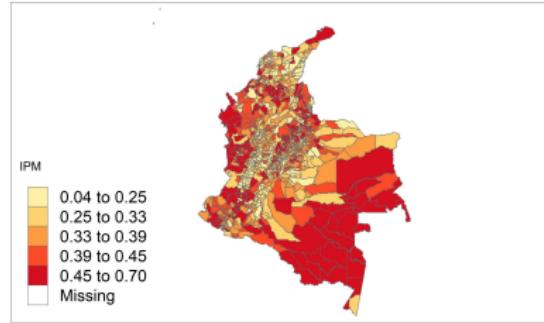
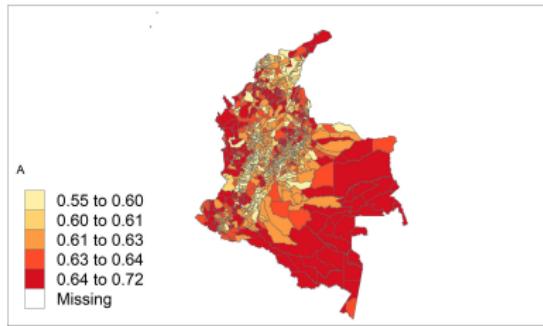
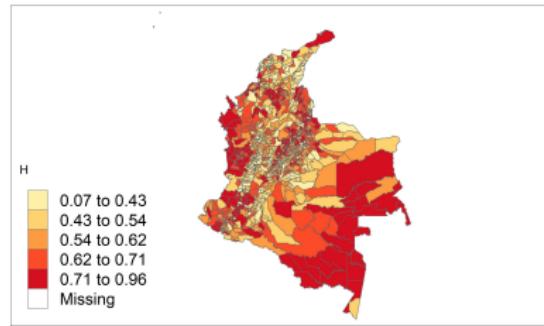


Figure 33: Componentes del IPM

Mapa de los privaciones que componen el IPM

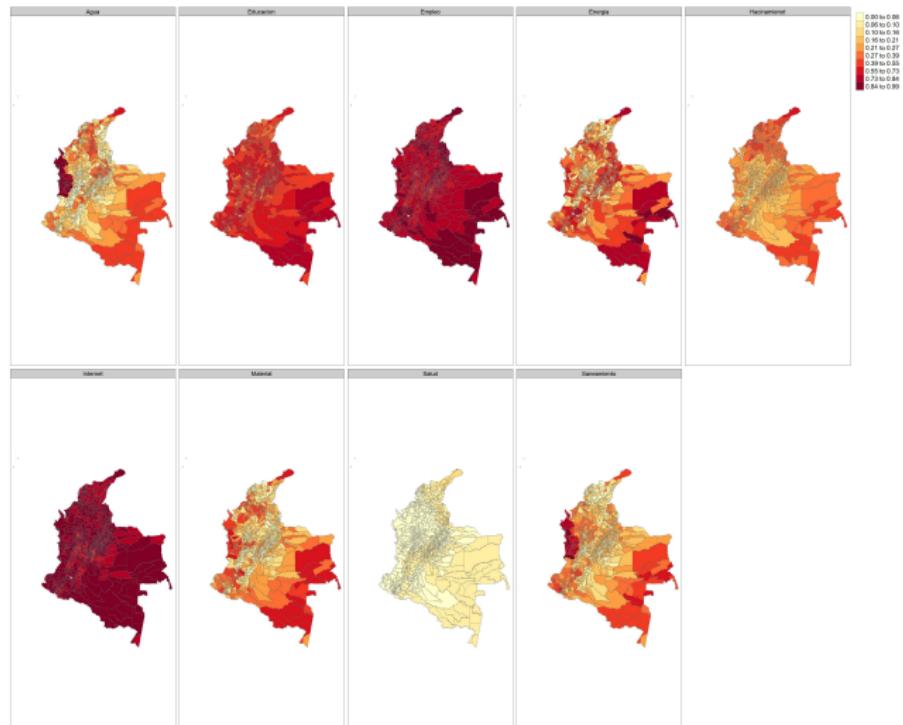


Figure 34: Privaciones del IPM

Modelo de área para estadísticas del mercado de trabajo

Definición del modelo multinomial

- ▶ Sea K el número de categorías de la variable de interés $Y \sim multinomial(\theta)$, con $\theta = (p_1, p_2, \dots, p_k)$ y $\sum_{k=1}^K p_k = 1$.
- ▶ Sea N_i el número de elementos en el i -ésimo dominio y N_{ik} el número de elementos que tienen la k -ésima categoría, note que $\sum_{k=1}^K N_{ik} = N_i$ y $p_{ik} = \frac{N_{ik}}{N_i}$.
- ▶ Sea \hat{p}_{ik} la estimación directa de p_{ik} y $v_{ik} = Var(\hat{p}_{ik})$ y denote el estimador de la varianza por $\hat{v}_{ik} = \widehat{Var}(\hat{p}_{ik})$

Consideraciones para el modelo multinomial.

El efecto diseño cambia entre categoría, por tanto, lo primero será definir el tamaño de muestra efectivo por categoría. Esto es:

La estimación de \tilde{n} esta dado por $\tilde{n}_{ik} = \frac{(\tilde{p}_{ik} \times (1 - \tilde{p}_{ik}))}{\hat{v}_{ik}}$,

$$\tilde{y}_{ik} = \tilde{n}_{ik} \times \hat{p}_{ik}$$

$$\text{luego, } \hat{n}_i = \sum_{k=1}^K \tilde{y}_{ik}$$

$$\text{de donde se sigue que } \hat{y}_{ik} = \hat{n}_i \times \hat{p}_{ik}$$

Modelo de área multinomial.

Sea $\theta = (p_1, p_2, p_3)^T = \left(\frac{N_{i1}}{N_i}, \frac{N_{i2}}{N_i}, \frac{N_{i3}}{N_i} \right)^T$, entonces el modelo multinomial para el i-ésimo dominio estaría dado por:

$$(\tilde{y}_{i1}, \tilde{y}_{i2}, \tilde{y}_{i3}) \mid \hat{n}_i, \theta_i \sim \text{multinomial}(\hat{n}_i, \theta_i)$$

Ahora, puede escribir p_{ik} como :

$$\ln\left(\frac{p_{i2}}{p_{i1}}\right) = X_i^T \beta_2 + u_{i2} \text{ y } \ln\left(\frac{p_{i3}}{p_{i1}}\right) = X_i^T \beta_3 + u_{i3}$$

Modelo de área multinomial.

Dada la restricción $1 = p_{i1} + p_{i2} + p_{i3}$ entonces

$$p_{i1} + p_{i1}(e^{X_i^T \beta_2} + u_{i2}) + p_{i1}(e^{X_i^T \beta_3} + u_{i3})$$

de donde se sigue que

$$p_{i1} = \frac{1}{1 + e^{X_i^T \beta_2} + u_{i2} + e^{X_i^T \beta_3} + u_{i3}}$$

Modelo de área multinomial.

Las expresiones para p_{i2} y p_{i3} estarían dadas por:

$$p_{i2} = \frac{e^{X_i^T \beta_2} + u_{i2}}{1 + e^{X_i^T \beta_2} + u_{i2} + e^{X_i^T \beta_3} + u_{i3}}$$

$$p_{i3} = \frac{e^{X_i^T \beta_3} + u_{i3}}{1 + e^{X_i^T \beta_2} + u_{i2} + e^{X_i^T \beta_3} + u_{i3}}$$

Estimación directa por municipio.

- ▶ Una persona encuestada puede tener uno de los siguientes estados: ocupadas, desocupadas o inactivo.
- ▶ Para cada dominio se calcula el número de personas ocupadas, desocupadas e inactivas en cada dominio y se estima la proporción de personas en cada una de estas categorías con sus respectivos errores estándar y efecto de diseño.

Selección de dominios

- ▶ Se emplean varias medidas de calidad, entre ellas, se cuenta el número de dominios que tienen dos o más unidades primarias de muestreo (UPM), así como el efecto de diseño mayor a 1 y las varianzas mayores a 0.
- ▶ Los dominios seleccionados son:
 - ▶ Dominios con dos o más upm.
 - ▶ Tener resultado en el DEFF.

El número de dominios seleccionados fue de 413

Modelo programando en STAN

Crear una función que simplifica los calculos.

```
functions {
    matrix pred_theta(matrix Xp, int p, matrix beta){
        int D1 = rows(Xp);
        real num1[D1, p];
        real den1[D1];
        matrix[D1,p] theta_p;
        for(d in 1:D1){
            num1[d, 1] = 1;
            num1[d, 2] = exp(Xp[d, ] * beta[1, ]' );
            num1[d, 3] = exp(Xp[d, ] * beta[2, ]' );
            den1[d] = sum(num1[d, ]);
        }
        for(d in 1:D1){
            for(i in 2:p){
                theta_p[d, i] = num1[d, i]/den1[d];
            }
            theta_p[d, 1] = 1/den1[d];
        }
    return theta_p ;
    }}}
```

Modelo programando en STAN

```
data {  
    int<lower=1> D; // número de dominios  
    int<lower=1> P; // categorías  
    int<lower=1> K; // cantidad de regresores  
    int y_tilde[D, P]; // matriz de datos  
    matrix[D, K] X_obs; // matriz de covariables  
    int<lower=1> D1; // número de dominios  
    matrix[D1, K] X_pred; // matriz de covariables  
}  
parameters {  
    matrix[P-1, K] beta;// matriz de parámetros  
    real<lower=0> sigma2_u1;  
    real<lower=0> sigma2_u2;  
    vector[D] u1;  
    vector[D] u2;  
}
```

Modelo programando en STAN

```
transformed parameters {
    simplex[P] theta[D];// vector de parámetros;
    real num[D, P];
    real den[D];
    real<lower=0> sigma_u1;
    real<lower=0> sigma_u2;
    sigma_u1 = sqrt(sigma2_u1);
    sigma_u2 = sqrt(sigma2_u2);
    for(d in 1:D){
        num[d, 1] = 1;
        num[d, 2] = exp(X_obs[d, ] * beta[1, ]' + u1[d]) ;
        num[d, 3] = exp(X_obs[d, ] * beta[2, ]' + u2[d]) ;
        den[d] = sum(num[d, ]);
    }
    for(d in 1:D){
        for(p in 2:P){
            theta[d, p] = num[d, p]/den[d];
        }
        theta[d, 1] = 1/den[d];
    }
}
```

Modelo programando en STAN

```
model {
    u1 ~ normal(0, sigma_u1);
    u2 ~ normal(0, sigma_u2);
    sigma2_u1 ~ inv_gamma(0.0001, 0.0001);
    sigma2_u2 ~ inv_gamma(0.0001, 0.0001);
    for(p in 2:P){
        for(k in 1:K){
            beta[p-1, k] ~ normal(0, 10000);
        }
    }
    for(d in 1:D){
        target += multinomial_lpmf(y_tilde[d, ] | theta[d, ]);
    }
}

generated quantities {
    matrix[D1,P] theta_pred;
    theta_pred = pred_theta(X_pred, P, beta);
}
```

Dominios observados

Seleccionar las variables del modelo y crear matriz de covariables.

```
names_cov <-  
  c("dam2", "tasa_desocupacion", "hacinamiento",  
    "piso_tierra", "luces_nocturnas",  
    "cubrimiento_cultivo", "modificacion_humana"  
  )  
X_pred <-  
  anti_join(  
    statelevel_predictors_df %>%  
    select(all_of(names_cov)),  
    indicador_dam1 %>% select(dam2))
```

Dominios NO observados

Creando la matriz de covariables para los dominios no observados (X_pred) y los observados (X_obs)

```
X_pred %<>%
  data.frame() %>%
  select(-dam2) %>% as.matrix()

X_obs <- inner_join(indicador_dam1 %>%
                      select(dam2, id_orden),
statelevel_predictors_df %>%
  select(all_of(names_cov))) %>%
  arrange(id_orden) %>%
  data.frame() %>%
  select(-dam2, -id_orden) %>%
  as.matrix()
```

Calculando el $n_{efectivo}$ y el \tilde{y}

```
D <- nrow(indicador_dam1)
P <- 3 # Ocupado, desocupado, inactivo.
Y_tilde <- matrix(NA, D, P)
n_tilde <- matrix(NA, D, P)
Y_hat <- matrix(NA, D, P)

# n efectivos ocupado
n_tilde[,1] <-
(indicador_dam1$Ocupado*(1 - indicador_dam1$Ocupado))/
indicador_dam1$Ocupado_var

Y_tilde[,1] <- n_tilde[,1]* indicador_dam1$Ocupado
```

Calculando el $n_{efectivo}$ y el \tilde{y}

```
n_tilde[,2] <-  
(indicador_dam1$Desocupado*(1 -  
indicador_dam1$Desocupado))/  
indicador_dam1$Desocupado_var  
Y_tilde[,2] <-  
n_tilde[,2]* indicador_dam1$Desocupado  
  
# n efectivos Inactivo  
n_tilde[,3] <-  
(indicador_dam1$Inactivo*(1 -  
indicador_dam1$Inactivo))/  
indicador_dam1$Inactivo_var  
  
Y_tilde[,3] <- n_tilde[,3]* indicador_dam1$Inactivo
```

Calculado \hat{Y} y \hat{n}_i

```
ni_hat = rowSums(Y_tilde)
Y_hat[,1] <- ni_hat* indicador_dam1$Ocupado
Y_hat[,2] <- ni_hat* indicador_dam1$Desocupado
Y_hat[,3] <- ni_hat* indicador_dam1$Inactivo
Y_hat <- ceiling(Y_hat)
```

Creando lista de datos para STAN

```
X1_obs <- cbind(matrix(1,nrow = D,ncol = 1),X_obs)
K = ncol(X1_obs)
D1 <- nrow(X_pred)
X1_pred <- cbind(matrix(1,nrow = D1,ncol = 1),X_pred)

sample_data <- list(D = D,
                      P = P,
                      K = K,
                      y_tilde = Y_hat,
                      X_obs = X1_obs,
                      X_pred = X1_pred,
                      D1 = D1)
```

Compilando el modelo en STAN

```
fit_mcmc2 <- stan(  
  file = "www/08_Mod_Trabajo/00_Multinomial_simple_no_cor.stan",  
  data = sample_data,  
  verbose = TRUE,  
  warmup = 1000, # number of warmup iterations per chain  
  iter = 2000,   # total number of iterations per chain  
  cores = 4,     # number of cores (could use one per chain)  
)  
  
saveRDS(fit_mcmc2,  
        "www/08_Mod_Trabajo/fit_multinomial_no_cor.Rds")
```

Validación del modelo

La validación de un modelo es esencial para evaluar su capacidad para predecir de manera precisa y confiable los resultados futuros. En el caso de un modelo de área con respuesta multinomial, la validación se enfoca en medir la precisión del modelo para predecir las diferentes categorías de respuesta.

Chequeo predictivo posterior

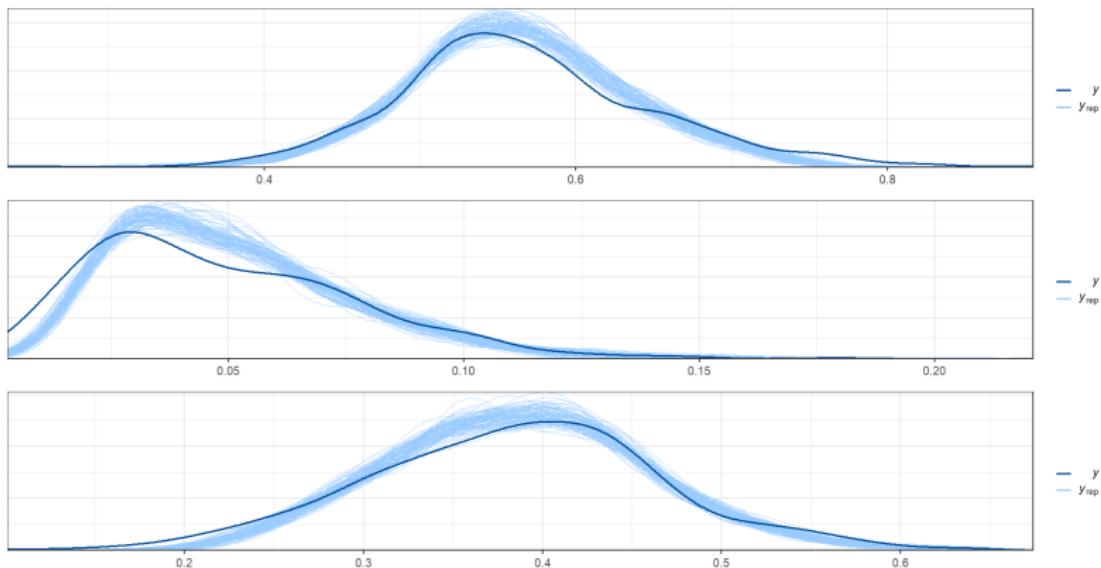


Figure 35: PPC modelo multinomial

Estimación de los parámetros.

Las estimaciones de los parámetros son obtenida directamente de las cadenas generadas por STAN, el proceso es organizar la información para tener concordancia con cada uno de los dominios. Luego de esto se obtiene la siguiente tabla.

Table 25: Estimación del modelo multinomial

dam2	Ocupado_mod	Desocupado_mod	Inactivo_mod
05001	0.5773	0.0776	0.3451
05002	0.4675	0.0308	0.5017
05031	0.5401	0.0577	0.4022
05034	0.5581	0.0311	0.4108
05045	0.5037	0.0773	0.4189

Metodología de Benchmarking

El proceso de Benchmarking se debe realizar para cada una de las categorías siguiendo los pasos que fueron realizados previamente. La distribución de los ponderadores se muestran a continuación.

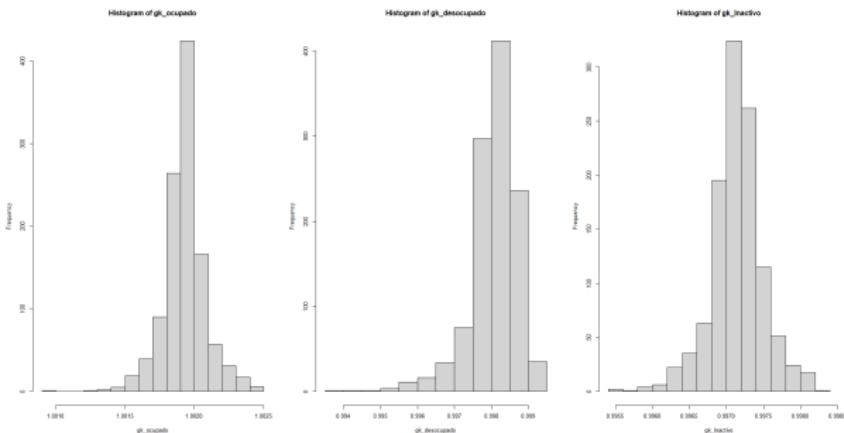


Figure 36: Distribución de los pesos

Mapas de los indicadores del mercado laboral

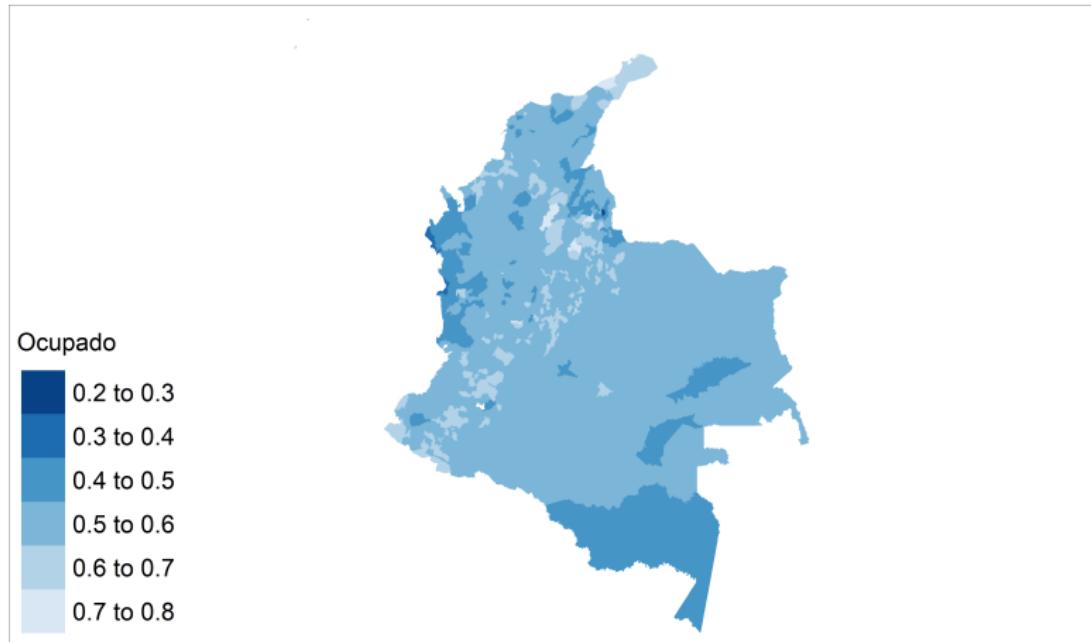


Figure 37: Mapas de Ocupados

Mapas de los indicadores del mercado laboral

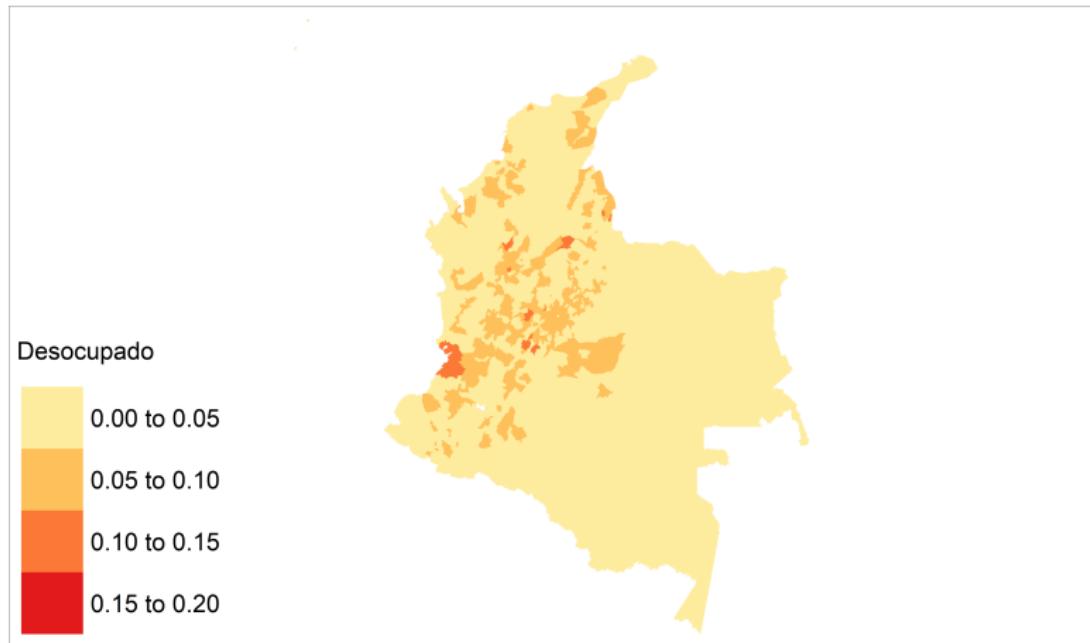


Figure 38: Mapas de Desocupados

Mapas de los indicadores del mercado laboral

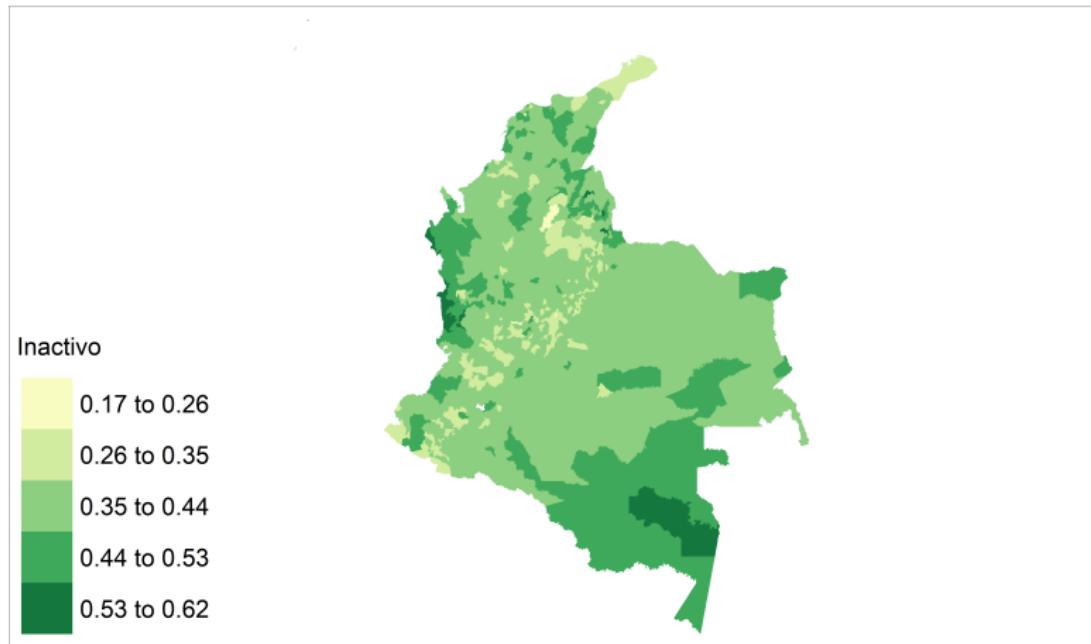


Figure 39: Mapas de Inactivo

¡Gracias!

Email: andres.gutierrez@cepal.org