**Ejercicio obligatorio 4 del seminario de imputación de datos perdidos**

El archivo *Multicento\_NJ\_perdidos.dat* tiene 2376 niños con déficit de aprendizaje de diferentes centros escolares de la ciudad de Nueva Jersey (edades de 11-13 años, 65% varones). Aproximadamente la mitad de ellos en cada centro recibieron un programa en fase experimental para fomentar sus habilidades cognitivas: el programa se hizo en horario extraescolar y se realizaron actividades para fomentar el aprendizaje de actividades cívicas, juegos matemáticos en grupos, videos y juegos para fomentar la creatividad, actividades sociales, etc. El programa se llevó a cabo en el curso 2010/11 monitorizado por el centro de investigación *Educational Testing Service* y subvencionado por la consejería de educación del distrito de Nueva Jersey*.*

La variable dependiente es el *rendimiento* escolar medida en una escala que va de 0 a 50 puntos que contempla no solo las notas medias, sino su involucración con otros compañeros, participación en clase, conductas proactivas, etc. La medida se registró al final del programa.

La variable de identificación de las unidades de nivel 2, el centro educativo, es *idcentro*.

La primera variable predictora es *programa* (0 = no aplicó y 1 = sí aplicó). Esta es una variable de nivel 1 puesto que varía *intracentro*. La segunda variable predictora de nivel 1 es el nivel de estudios de los padres de 1 (estudios básicos) a 4 (estudios superiores) llamada *est*.

Hay una variable predictora de nivel 2 llamada *fondos* (en miles de dólares que registró las ayudas recibidas por el centro escolar subvencionadas públicamente por el distrito de Nueva Jersey) (p.ej., el valor 4,5 son 4.500$ que el centro recibe anualmente en ayudas).

El objetivo de los investigadores es, por un lado, valorar si el programa ha tenido éxito a la hora de mejorar el rendimiento de los estudiantes y conocer cómo la variable *fondos* modera (si lo hace) la relación entre el programa y el rendimiento. Secundariamente se quiere valorar si el nivel educativo de los padres se relaciona positivamente con el rendimiento.

Existe pérdida de datos en la variable dependiente *rendimiento* (28,5%) y en los estudios de los padres (23,9).

Necesitas cargar las librerías:

library(mice) # -- libreria para imputacion multiple con ecuaciones encadenadas

library(miceadds) # -- incorporar mas metodos a mice

library(micemd) # -- incorporar mas metodos a mice

library(mitml) # -- libreria para trabajar con valores perdidos

library(lme4) # -- libreria estimar modelos multinivel

library(lmerTest) # -- para obtener los p-values de los efectos en multinivel

1. Estima primero con un método de **eliminación por lista** (con la función *lmer()* que usa el paquete lme4) el coeficiente de correlación intraclase (ICC). Nota: Conviene que crees antes un archivo sin valores perdidos con la función *na.omit(dataframe)* para responder a las preguntas 1, 2, 3 y 4.

**¿Crees conveniente abordar estos datos con un modelo multinivel? ¿Por qué?**

El valor del coeficiente de correlación intraclase (ICC) obtenido es de **0.299**, lo cual indica que aproximadamente el **30% de la varianza total del rendimiento escolar** se debe a diferencias **entre centros educativos** (nivel 2), mientras que el 70% restante se debe a diferencias **entre estudiantes dentro de cada centro** (nivel 1). Debido a esto, se observa conveniente incorporar a los centros educativos dentro del modelo en una estructura jerárquica o multinivel.x

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

2. Incorpora ahora estos efectos fijos en el modelo siguiendo con lmer()

programa + fondos + programa:fondos + est

Continuaremos en un modelo de intersecciones aleatorias: esto es, la sintaxis de la estructura aleatoria es (1|centro)

**¿Cuáles de los efectos fijos son significativos? ¿Y cuáles no?**

|  |
| --- |
|  |
| Parameter | **Coefficient** | **SE** | **95% CI** | **t(1285)** | **p** |
| *Fixed Effects* | | | | | |
| (Intercept) | 5.18 | 1.15 | (2.92, 7.44) | 4.50 | < .001 |
| programa | 7.83 | 0.53 | (6.80, 8.87) | 14.82 | < .001 |
| fondos | 2.50 | 0.49 | (1.54, 3.47) | 5.10 | < .001 |
| est | -0.09 | 0.08 | (-0.24, 0.07) | -1.10 | 0.272 |
| programa × fondos | 1.20 | 0.23 | (0.76, 1.65) | 5.28 | < .001 |
| *Random Effects* | | | | | |
| SD (Intercept: idcentro) | 2.72 |  |  |  |  |
| SD (Residual) | 2.97 |  |  |  |  |

Los efectos fijos de programa (β = 7.83, p < .001) y fondos (β = 2.50, p < .001) son **estadísticamente significativos**, lo que indica que cada uno de estos factores tiene una contribución relevante en la predicción del rendimiento escolar. En contraste, el nivel educativo de los padres (est) no mostró un efecto significativo (β = -0.09, p = .272), por lo que no habría evidencia estadística de que esta variable se asocie de manera directa con el rendimiento de los estudiantes, una vez controlados los demás factores.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3. Interpreta el efecto de la interacción programa:fondos? **¿Cómo modera la variable de nivel 2 *fondos* la relación que hay entre el *programa* y el *rendimiento*? ¿Cómo afectaría el programa por cada unidad que aumenta la variable *fondos*?**

La interacción programa × fondos (β = 1.20, p < .001) se observa como estadísticamente significativa moderando positivamente la efectividad del programa. Esto es que **por cada unidad adicional en la variable fondos (es decir, por cada $1 000 dólares extra recibidos por el centro educativo),** el **efecto del programa sobre el rendimiento escolar aumenta en 1.20 puntos**. Dicho de otra forma**, cuanto más fondos tiene un centro, mayor es el impacto del programa sobre el rendimiento estudiantil.**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

4. Incluye un modelo con pendientes aleatorias para la variable *programa* para terminar con lmer. **¿Crees que ajusta el modelo de pendientes aleatorias mejor que el que incluye únicamente intersecciones aleatorias?** Puedes servirte de la función anova(modelo1, modelo2) para averiguar esto.

El modelo que incorpora pendientes aleatorias para el predictor programa muestra un **ajuste significativamente mejor** que el modelo que incluye únicamente intersecciones aleatorias (χ²(2) = 32.73, p < .001). Esto indica que el **efecto del programa sobre el rendimiento varía entre los centros escolares**, es decir, **no es constante en todas las unidades de nivel 2.** Incorporar pendientes aleatorias permite modelar esta variabilidad contextual, generando estimaciones más precisas y realistas de los efectos del programa.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Interpreta qué significa incorporar pendientes aleatorias para el predictor *programa*.**

Incorporar pendientes aleatorias para el predictor programa significa que no se asume que el efecto del programa sobre el rendimiento escolar sea igual en todos los centros educativos (nivel 2), sino que se permite que dicho efecto varíe de un centro a otro. Esto implica que cada centro tiene su propia pendiente asociada al efecto del programa, es decir, la magnitud con la que el programa influye en el rendimiento puede aumentar o disminuir dependiendo del centro.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Por último, estimaremos el modelo con mice (imputación múltiple). Asegúrate de que vas a utilizar el archivo original con datos observados y perdidos. Imputaremos los valores perdidos de las variables *rendimiento* y *est* (recuerda que son las únicas variables con pérdida de datos). Es importante que imputes estas dos variables con la interacción programa\*fondos, ya que es un término esencial de tu modelo sustantivo, de forma que tendrás que crearlo en el dataframe primero.

Para la variable *rendimiento* (cuantitativa) usaremos el método de imputación *2l.glm.norm* y para la variable *est* (ordinal) el método de imputación *2l.pmm*. A continuación, debes parametrizar el objeto *predictorMatrix*. Recuerda que la variable de identificación de las unidades de nivel 2 tienen en esa matriz el código -2. Luego, puedes optar porque los predictores de nivel 1 (es decir, *rendimiento*, *programa* y *est*) mantengan un valor de 1 (estimación solo con efectos fijos) o modificar esto a un valor 2 (estimación que añade pendientes aleatorias). La recomendación es hacer una prueba de sensibilidad para ver si afecta cómo la estructura aleatoria afecta a los resultados, pero de optar por uno de los dos métodos **¿cuál crees que es recomendable?**

Se imputaron 10 bases con dos configuraciones del predictorMatrix. La versión A incorporó solo efectos fijos (pred = 1) y la versión B permitió pendientes aleatorias para los predictores de nivel 1 (pred = 2). Al comparar los modelos de análisis posteriores, los coeficientes y sus niveles de significación se mantuvieron prácticamente inalterados. No obstante, la versión con pendientes aleatorias produjo errores estándar ligeramente más grandes, reflejando mayor varianza entre centros. Dado que nuestro modelo sustantivo final incluye variación aleatoria únicamente en la pendiente de `programa`, y la imputación con pendientes aleatorias es más general respecto a ese modelo, adoptamos esta última como la opción preferente.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Tras imputar (puedes optar por poner bases de datos, tardará unos minutos) ajusta el modelo de pendientes aleatorias: lmer(rendimiento ~ programa + fondos + programa:fondos + est + (programa|idcentro).

**Compara los resultados de *mice()* y el que obtenías con *lmer()* en términos de la significación estadística de los predictores, del (posible) sesgo y de la eficiencia de los estimadores**.

|  |  |  |  |
| --- | --- | --- | --- |
| Predicción | Casos completos (sin NA) | Imputación múltiple (m = 10; mice, pred = 2) | Cambio / interpretación |
| Intercepto | ≈ 5.10 ± 1.10 (SE) | 5.07 ± 1.13 | Sin cambios apreciables |
| programa | 7.88 ± 0.78 | 7.57 ± 0.74 | Ligera reducción (≈ –0.3 ptos) y SE similar. El análisis con la imputación corrige el sesgo de haber analizado solo los casos con rendimiento observado. |
| fondos | 2.56 ± 0.47 | 2.47 ± 0.47 | Sin cambios apreciables |
| programa × fondos | 1.18 ± 0.34 | 1.35 ± 0.32 | Aumenta ligeramente (+0.17) y el SE desciende un poco. La interacción sigue con significatividad estadística. |
| est | –0.10 ± 0.08 (ns) | –0.055 ± 0.101 (ns) | El efecto continúa irrelevante. El mayor SE refleja la incertidumbre añadida por imputar un 24 % de valores faltantes en est. |
| Var (pendiente programa) | ≈ 2.29 | 2.17 | Misma magnitud; se confirma que la efectividad del programa varía entre centros. |
| ICC | 0.454 | 0.453 | Sin cambios. |

Todos los predictores que eran significativos en la base completa **(programa, fondos, programa × fondos) siguen siéndolo** trasla imputación. El predictor **est** permanece no significativo. Se observa que con un 29 % de pérdida en **rendimiento** y 24 % en **est**, la eliminación por lista puede sesgar los coeficientes, aunque en estos análisis se aprecian mínimos. En general, la imputación múltiple con pendientes aleatorias preserva la estructura jerárquica y confirma las conclusiones obtenidas con casos completos.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***Anexo Código R***

##%######################################################%##  
# #  
#### Ejercicio 4 ###  
#### Brian Norman Peña Calero ###  
#### Seminario de Valores Perdidos ###  
# #  
##%######################################################%##  
  
# Carga de Paquetes -------------------------------------------------------  
  
library(tidyverse)  
# Librerías principales  
library(mice) # Imputación múltiple con ecuaciones encadenadas  
library(lme4) # Modelos multinivel  
library(lmerTest) # p-values para lme4  
library(parameters) # Resultados formateados  
library(performance) # ICC y evaluación de modelos  
library(miceadds) # Métodos adicionales para mice  
library(micemd) # Métodos adicionales para imputación multinivel  
library(mitml) # Para combinar resultados de imputaciones múltiples  
  
  
# Importar Datos Ejercicio ----------------------------------------------  
  
multicentro <- read\_delim("Multicento\_NJ\_perdidos.dat")  
multicentro  
  
# Revisar el patrón de valores perdidos  
summary(multicentro)  
md.pattern(multicentro)  
  
# Guardar solo completos  
multicentro\_complete <- na.omit(multicentro)  
  
  
# Pregunta 1 --------------------------------------------------------------  
  
# Modelo nulo (solo intercepto aleatorio por centro)  
modelo\_nulo <- lmer(rendimiento ~ (1 | idcentro),   
 data = multicentro\_complete, REML = FALSE)  
  
# Resumen del modelo  
summary(modelo\_nulo)  
parameters(modelo\_nulo)  
  
# Calcular el coeficiente de correlación intraclase (ICC)  
icc(modelo\_nulo)  
  
  
# Pregunta 2 y 3 ------------------------------------------------------------  
  
# Ajustar el modelo con efectos fijos y estructura aleatoria  
modelo\_fijo <- lmer(  
 rendimiento ~ programa + fondos + programa:fondos + est + (1 | idcentro),  
 data = multicentro\_complete,  
 REML = FALSE  
)  
  
# Ver resumen del modelo  
summary(modelo\_fijo)  
parameters(modelo\_fijo)  
  
  
# Pregunta 4 --------------------------------------------------------------  
  
# Modelo con pendientes e intersecciones aleatorias para "programa"  
modelo\_pendiente <- lmer(  
 rendimiento ~ programa + fondos + programa:fondos + est + (programa | idcentro),  
 data = multicentro\_complete,  
 REML = FALSE  
)  
  
# Resumen del modelo  
summary(modelo\_pendiente)  
parameters(modelo\_pendiente)  
  
# Comparar modelos (intersecciones vs pendientes e intersecciones)  
anova(modelo\_fijo, modelo\_pendiente)  
  
  
# Pregunta 5 --------------------------------------------------------------  
  
# Volver al dataframe original con NA  
multicentro\_na <- multicentro  
  
# Crear la variable de interacción  
multicentro\_na$programa\_fondos <- multicentro\_na$programa \* multicentro\_na$fondos  
  
# Función de imputación personalizada  
imputar\_modelo <- function(pred\_valor = 1, seed = 123) {  
 ini <- mice(multicentro\_na, maxit = 0)  
 pred <- ini$predictorMatrix  
 meth <- ini$method  
   
 # Identificación de nivel 2  
 pred[,"idcentro"] <- -2  
 pred["idcentro", ] <- 0  
   
 # Solo imputamos rendimiento y est  
 meth[] <- ""  
 meth["rendimiento"] <- "2l.glm.norm"  
 meth["est"] <- "2l.pmm"  
   
 # Configuración general  
 pred["rendimiento", "rendimiento"] <- 0  
 pred["est", "est"] <- 0  
 pred["programa\_fondos", ] <- 0  
 pred[,"programa\_fondos"] <- 1  
   
 # Ajustamos los valores de efectos fijos o aleatorios  
 pred["rendimiento", c("programa", "est")] <- pred\_valor  
 pred["est", c("programa", "rendimiento")] <- pred\_valor  
   
 # Imputación  
 set.seed(seed)  
 mice(multicentro\_na, m = 10, predictorMatrix = pred, method = meth)  
}  
  
# A: Solo efectos fijos  
imp\_fijo <- imputar\_modelo(pred\_valor = 1)  
  
# B: Con pendientes aleatorias  
imp\_aleatorio <- imputar\_modelo(pred\_valor = 2)  
  
  
# Convertir a listas mitml  
imp\_fijo\_list <- mids2mitml.list(imp\_fijo)  
imp\_aleatorio\_list <- mids2mitml.list(imp\_aleatorio)  
  
# Ajustar modelos con la fórmula definida dentro de with()  
fit\_fijo <- with(imp\_fijo\_list, {  
 lmer(rendimiento ~ programa + fondos + programa\_fondos + est + (1 | idcentro))  
})  
  
fit\_aleatorio <- with(imp\_aleatorio\_list, {  
 lmer(rendimiento ~ programa + fondos + programa\_fondos + est + (programa | idcentro))  
})  
  
  
# Comparar estimaciones  
res\_fijo <- testEstimates(fit\_fijo, extra.pars = TRUE)  
res\_aleatorio <- testEstimates(fit\_aleatorio, extra.pars = TRUE)