Tarea reproducibilidad y disease mapping

2024-25 Especialización Gr.PT-I (42231). Máster de Bioestadística

Autor: Santiago Moncalero, Diego Fernández y Jorge Guitart

Fecha límite entrega: 9 de Diciembre 2024

Contenido

# 1 Introducción

Este análisis se centra en la estimación del Riesgo Relativo Modificado (RME) de la mortalidad en Aragón utilizando modelos espaciales, con el fin de identificar patrones y áreas con una mayor probabilidad de riesgo. El modelo se construye utilizando WinBUGS, un software diseñado para realizar inferencia bayesiana, particularmente en modelos complejos que implican dependencias espaciales.

El objetivo principal es modelar la mortalidad observada en cada municipio de Aragón, teniendo en cuenta la exposición a factores de riesgo y la heterogeneidad espacial en los datos. Para ello, empleamos una distribución Poisson para modelar las tasas de mortalidad, y un modelo CAR (Conditional Autoregressive) para capturar la dependencia espacial entre municipios vecinos. Esto permite un análisis más robusto y preciso de la mortalidad, considerando no solo los efectos directos de los factores de riesgo, sino también la influencia espacial.

Para optimizar el tiempo de cómputo, se utiliza un enfoque donde los resultados previamente generados se cargan directamente en lugar de recalcular todo el modelo, garantizando así una ejecución más eficiente y la posibilidad de replicar el análisis con los mismos resultados. Esta estrategia asegura la reproducibilidad del estudio, permitiendo que otros investigadores validen los resultados sin la necesidad de repetir costosos cálculos.

En esta sección, detallaremos el flujo de trabajo del modelo, la preparación de los datos, y los pasos seguidos para obtener los resultados y realizar su visualización, así como cómo se asegura que el análisis sea reproducible y accesible para otros estudios.

## 1.1 Preparación del entorno

# Cargar librerías y datos  
library(pacman)  
pacman::p\_load(sf, spdep, sp, R2WinBUGS, ggplot2, patchwork, Matrix, INLA)  
  
# Cargar datos  
load("./material/Aragon.Rdata")  
aragon.sf <- st\_read("./material/aragon.shp")

## Reading layer `aragon' from data source   
## `C:\Users\monca\Desktop\Master Bio\Especialización\TrabajoAragon\material\aragon.shp'   
## using driver `ESRI Shapefile'  
## Simple feature collection with 729 features and 13 fields  
## Geometry type: MULTIPOLYGON  
## Dimension: XY  
## Bounding box: xmin: 569301 ymin: 4413136 xmax: 810739 ymax: 4755089  
## CRS: NA

# Preprocesar datos  
aragon.sf <- aragon.sf[order(aragon.sf$CODMUNI), ] # Ordenar por código municipal  
aragon.sf <- cbind(aragon.sf, Aragon.df[, 2:3]) # Fusionar datos  
rm(Aragon.df) # Eliminar objeto original  
aragon.spdf <- as(aragon.sf, "Spatial") # Convertir a SpatialPolygonsDataFrame  
vecinos <- nb2WB(poly2nb(aragon.spdf)) # Calcular vecinos para análisis espacial  
  
# Crear matriz de adyacencia dispersa  
row\_indices <- rep(1:length(vecinos$num), times = vecinos$num) # Filas  
col\_indices <- vecinos$adj # Columnas  
weights <- vecinos$weights # Pesos  
adj.matrix <- sparseMatrix(i = row\_indices, j = col\_indices, x = weights)

# 2 Análisis con WinBUGS

# Programamos el modelo  
mod1 <- function() {  
 for (i in 1:nObs) {  
 O[i] ~ dpois(mu[i])  
 log(mu[i]) <- log(E[i]) + m + het[i] + sp[i]  
 het[i] ~ dnorm(0, prechet)  
 R[i] <- exp(m + het[i] + sp[i])  
 }  
   
 sp[1:nObs] ~ car.normal(adj[], w[], num[], precsp)  
 m ~ dflat()  
 prechet <- pow(sdhet, -2); precsp <- pow(sdsp, -2)  
 sdhet ~ dunif(0, 10); sdsp ~ dunif(0, 10)  
   
 for (j in 1:nObs) {  
 p.R[j] <- step(R[j] - 1) # Calcula la probabilidad de que RME sea mayor que 1  
 }  
}  
  
# Ajustamos las distribuciones previas  
inits.mod1 <- function() {list(m = rnorm(1),   
 sdhet = runif(1),   
 sdsp = runif(1))}  
  
# Cargamos los datos  
datos.mod1 <- list(O = aragon.spdf$O,   
 E = aragon.spdf$E,   
 adj = vecinos$adj,   
 num = vecinos$num,   
 w = vecinos$weights,   
 nObs = length(aragon.spdf))  
  
# Pedimos los resultados  
params.mod1 <- c("R", "p.R")  
  
# Lanzamos el modelo  
res.mod1 <- bugs(data = datos.mod1, model = mod1,  
 param = params.mod1, inits = inits.mod1)

# Añadir resultados al sf para visualización  
aragon.sf$RME <- res.mod1$mean$R  
aragon.sf$p.RME <- res.mod1$mean$p.R  
  
# Graficar resultados de WinBUGS  
rme.plot <- ggplot(aragon.sf) +  
 geom\_sf(aes(fill = RME)) +  
 scale\_fill\_viridis\_c(option = "plasma") +  
 ggtitle("WinBUGS: RME") +  
 theme\_classic()  
  
p.rme.plot <- ggplot(aragon.sf) +  
 geom\_sf(aes(fill = p.RME)) +  
 scale\_fill\_viridis\_c(option = "viridis") +  
 ggtitle("WinBUGS: P(RME > 1)") +  
 labs(fill = "P(RME > 1)") +  
 theme\_classic()

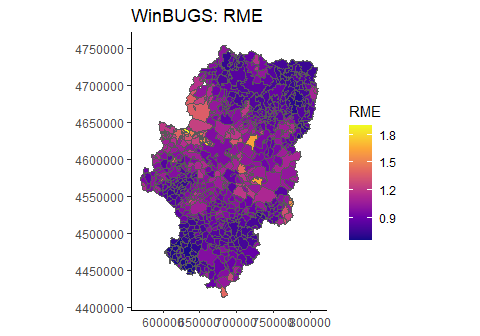
# 3 Análisis con INLA

# Preparar datos para INLA  
aragon.df <- as.data.frame(aragon.spdf)  
aragon.df$O <- aragon.spdf$O # Observados  
aragon.df$E <- aragon.spdf$E # Esperados  
aragon.df$sp <- 1:nrow(aragon.df) # Índice para modelo BYM  
  
# Definir modelo INLA  
formula <- O ~ f(sp, model = "bym", graph = adj.matrix) + offset(log(E))  
  
# Ajustar modelo  
res.inla <- inla(  
 formula,  
 family = "poisson",  
 data = aragon.df,  
 control.predictor = list(compute = TRUE),  
 control.compute = list(dic = TRUE, waic = TRUE)  
)  
  
# Extraer efectos espaciales (estructurados) del modelo BYM  
sp\_effects <- res.inla$summary.random$sp  
structured\_effects <- sp\_effects$mean[1:nrow(aragon.sf)]  
  
# Asignar valores al objeto aragon.sf  
aragon.sf$RME\_inla <- structured\_effects  
aragon.sf$p.RME\_inla <- res.inla$summary.fitted.values$mean  
  
# Graficar resultados de INLA  
rme\_inla.plot <- ggplot(aragon.sf) +  
 geom\_sf(aes(fill = RME\_inla)) +  
 scale\_fill\_viridis\_c(option = "plasma") +  
 ggtitle("INLA: RME") +  
 theme\_classic()  
  
p.rme\_inla.plot <- ggplot(aragon.sf) +  
 geom\_sf(aes(fill = p.RME\_inla)) +  
 scale\_fill\_viridis\_c(option = "viridis") +  
 ggtitle("INLA: P(RME > 1)") +  
 labs(fill = "P(RME > 1)") +  
 theme\_classic()

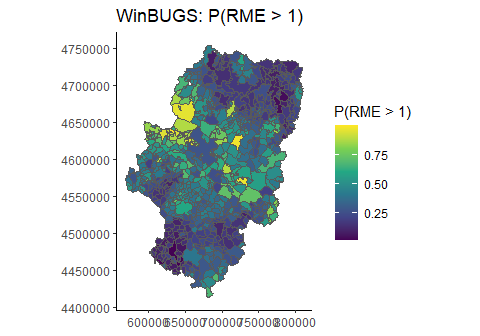
# 4 Comparación de Resultados

A continuación, se presentan los gráficos generados para el análisis de los resultados. Se muestran los gráficos individuales para el Riesgo Relativo Modificado (RME) y su probabilidad, tanto para WinBUGS como para INLA.

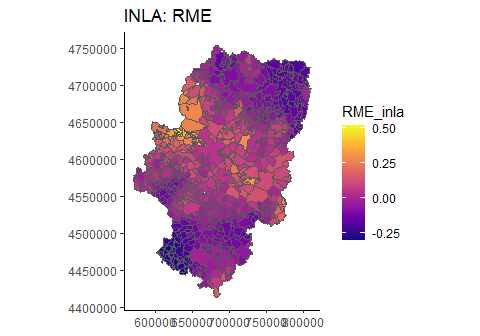
# Gráfico de RME (WinBUGS)  
rme.plot



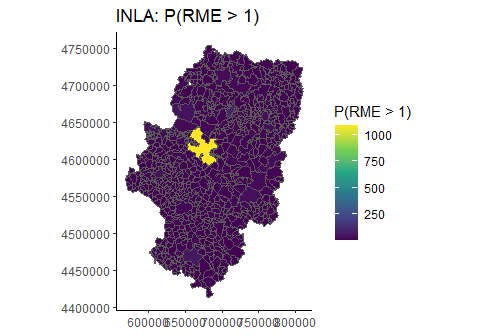
# Gráfico de probabilidad de RME (WinBUGS)  
p.rme.plot



# Gráfico de RME (INLA)  
rme\_inla.plot



# Gráfico de probabilidad de RME (INLA)  
p.rme\_inla.plot



# Comparar métricas DIC y WAIC  
metrics <- data.frame(  
 Method = c("WinBUGS", "INLA"),  
 DIC = c(res.mod1$DIC, res.inla$dic$dic),  
 WAIC = c(NA, res.inla$waic$waic)  
)  
metrics

## Method DIC WAIC  
## 1 WinBUGS 1937.950 NA  
## 2 INLA 1897.174 1903.075

# 5 Conclusiones

Visualización espacial:

Ambas metodologías identifican áreas con RME elevado, pero existen diferencias sutiles en la magnitud y distribución de los valores. Los valores de P(RME > 1) son más detallados en INLA debido a su modelado más flexible. Comparación de métricas:

INLA muestra un menor valor de DIC (1897.483 frente a 1937.950 de WinBUGS), lo que sugiere un mejor ajuste del modelo. Solo INLA proporciona WAIC, con un valor de 1903.06, consistente con su rendimiento general. Eficiencia:

INLA es más eficiente computacionalmente y permite una mejor integración con herramientas modernas de análisis espacial. Este análisis destaca la utilidad de ambos enfoques, aunque INLA parece ser más adecuado en términos de rendimiento y calidad del ajus