Actividad Final

Ejercicio de analisis de amenidades del Carchi

Adrián Enríquez

2024-09-01

# Parte 1: Indentificación y reflexión sobre las Amenidades de Salud

Se procede a realizar la busquedad de hospitales y clinicas en los cantones Espejo y Mira pertenecientes a la provincia del Carchi usando OpenStreetMap.

#Librerías  
library(osmdata)

Data (c) OpenStreetMap contributors, ODbL 1.0. https://www.openstreetmap.org/copyright

library(sf)

Warning: package 'sf' was built under R version 4.4.1

Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf\_use\_s2() is TRUE

library(tidyverse)

── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──  
✔ dplyr 1.1.4 ✔ readr 2.1.5  
✔ forcats 1.0.0 ✔ stringr 1.5.1  
✔ ggplot2 3.5.1 ✔ tibble 3.2.1  
✔ lubridate 1.9.3 ✔ tidyr 1.3.1  
✔ purrr 1.0.2

── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
✖ dplyr::filter() masks stats::filter()  
✖ dplyr::lag() masks stats::lag()  
ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors

library(readxl)  
library(spdep)

Warning: package 'spdep' was built under R version 4.4.1

Loading required package: spData

Warning: package 'spData' was built under R version 4.4.1

To access larger datasets in this package, install the spDataLarge  
package with: `install.packages('spDataLarge',  
repos='https://nowosad.github.io/drat/', type='source')`

library(lattice)

#Vector de parroquias urbanas y rurales del cantón Espejo y Mira  
  
  
parroquias\_espejo <- c( #Espejo  
 "El Ángel","27 De Septiembre", "El Goaltal",  
 "La Libertad", "San Isidro")  
 #Mira  
parroquias\_mira <- c("Mira","Concepción", "Jijón y Caamaño", "Juan Montalvo")  
  
parroquias\_total <- c(  
 "El Ángel","27 De Septiembre", "El Goaltal",  
 "La Libertad", "San Isidro",  
 "Mira","Concepción", "Jijón y Caamaño", "Juan Montalvo")  
#amenidades  
localizaciones <- c("hospital", "clinic")  
  
#Función para obtener los puntos usando una lista de parroquias del cantón Espejo y Mira  
get\_puntos <- function(zonas, provincia, pais, localizaciones) {  
   
 # lista para almacenar los resultados  
 puntos\_lista <- vector("list", length(zonas))  
   
 # Bucle por cada zona (parroquia)  
 for (i in seq\_along(zonas)) {  
 zona <- zonas[i]  
   
 # Obtener el bounding box para cada zona  
 bbox <- getbb(paste(zona, provincia, pais, sep = ", "))  
   
 # Verificar si el bbox es válido  
 if (is.null(bbox)) next  
   
 # Construir la consulta OSM y obtener los puntos  
 query <- opq(bbox) %>%   
 add\_osm\_feature(key = "amenity", value = localizaciones) %>%   
 osmdata\_sf()  
   
 # Si hay puntos disponibles, almacenarlos en la lista  
 if (!is.null(query$osm\_points) && nrow(query$osm\_points) > 0) {  
 puntos\_lista[[i]] <- query$osm\_points[, c("osm\_id", "geometry")]  
 }  
 }  
   
 # Combinar los resultados no nulos en un solo objeto sf  
 puntos\_sf <- do.call(rbind, puntos\_lista)  
   
 # Retornar el objeto sf con geometría  
 return(st\_as\_sf(puntos\_sf, crs = 4326))  
}  
#Guardo puntos  
puntos\_espejo <- get\_puntos(parroquias\_espejo,"Carchi", "Ecuador", localizaciones)  
puntos\_mira <- get\_puntos(parroquias\_mira,"Carchi", "Ecuador", localizaciones)  
puntos\_global <- get\_puntos(parroquias\_total,"Carchi", "Ecuador", localizaciones)  
  
#Numero de amenidades de hospitales de los cantones espejo y Mira  
nrow(puntos\_espejo)

[1] 19

nrow(puntos\_mira)

[1] 41

nrow(puntos\_espejo) + nrow(puntos\_mira)

[1] 60

En un inicio se iba a tomar como area de estudio a la parroquia de El Ángel perteneciente al cantón Espejo para analizar la cantidad de amenidades, pero al ser una parroquia pequeña no obtenía una cantidad de datos suficiente. Por lo tando, se tomó los puntos de cada parroquia de los cantones Espejo con 19 amenidades y Mira con 41 amenidades pertenecientes a la provincia de Carchi teniendo un total de 60 amenidades. Para ello, se usó OpenStreetMap para determinar las amenidades de la parroquias mencionadas en la variable parroquias agregando condiciones en el bbox para que se encuentre limitado de forma correcta (tome las parroquias dentro de los cantones en los que se basa el estudio).

# Parte 2: Unión de los polígonos de Zonas Censales a Nivel de Parroquias

Obtenemos la zona geometria de la provincia del carchi

# Lista las capas disponibles dentro del archivo GeoDatabase para identificar la que necesitamos.  
st\_layers("data/GEODATABASE\_NACIONAL\_2021/GEODATABASE\_NACIONAL\_2021.gdb")

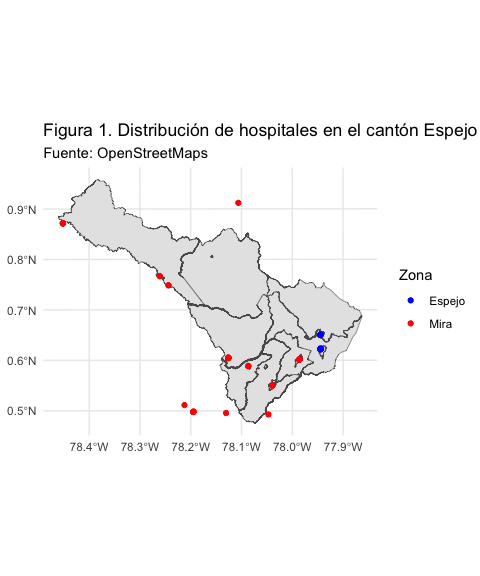
Driver: OpenFileGDB   
Available layers:  
 layer\_name geometry\_type features fields crs\_name  
1 aream\_a Multi Polygon 1847 8 WGS 84 / UTM zone 17S  
2 ca04\_a Multi Polygon 3951938 9 WGS 84 / UTM zone 17S  
3 ejes\_l Multi Line String 546887 9 WGS 84 / UTM zone 17S  
4 ingresos\_l Multi Line String 4153735 5 WGS 84 / UTM zone 17S  
5 loc\_p Point 59880 5 WGS 84 / UTM zone 17S  
6 man\_a Multi Polygon 234172 11 WGS 84 / UTM zone 17S  
7 sec\_a Multi Polygon 52898 7 WGS 84 / UTM zone 17S  
8 viv\_p Multi Point 5578282 9 WGS 84 / UTM zone 17S  
9 zon\_a Multi Polygon 5888 6 WGS 84 / UTM zone 17S

# Lee la capa "zon\_a" desde el archivo GeoDatabase y la guarda en el objeto 'zonas'.  
zonas <- st\_read("data/GEODATABASE\_NACIONAL\_2021/GEODATABASE\_NACIONAL\_2021.gdb",  
 layer = "zon\_a")

Reading layer `zon\_a' from data source   
 `/Users/adrianenriquez/Documents/MyProjects/R/Tarea\_Final/Analisis\_espacial\_amenidades\_salud\_Espejo\_Mira/data/GEODATABASE\_NACIONAL\_2021/GEODATABASE\_NACIONAL\_2021.gdb'   
 using driver `OpenFileGDB'  
Simple feature collection with 5888 features and 6 fields  
Geometry type: MULTIPOLYGON  
Dimension: XY  
Bounding box: xmin: -685704.9 ymin: 9445216 xmax: 1147852 ymax: 10162550  
Projected CRS: WGS 84 / UTM zone 17S

# Filtra las zonas para mantener solo aquellas cuyo código comienza con "0403" (Carchi, Espejo).  
zonas <- zonas %>%   
 filter(str\_detect(zon, "^0403|^0404"))

Para graficar empleamos el código del gráfico **ggplot2** y jugamos con las opciones de Quarto chunks



Lectura del archivo con la población del Ecuador por parroquias

# Lista las hojas disponibles en el archivo Excel para identificar la que necesitamos.  
excel\_sheets("data/01\_2022\_CPV\_Estructura\_poblacional.xlsx")

[1] "Índice" "1" "1.1" "1.2" "2" "2.1" "3" "4"   
 [9] "4.1" "4.2" "5" "5.1" "5.2"

# Lectura de la hoja "1.2" del archivo Excel, en el rango específico de celdas B10:G1299,   
poblacion <- read\_excel("data/01\_2022\_CPV\_Estructura\_poblacional.xlsx",   
 sheet = "1.2",  
 range = "B10:G1299"  
 )

New names:  
• `` -> `...1`  
• `` -> `...2`  
• `` -> `...3`  
• `` -> `...4`

# Renombra las columnas del dataframe para hacerlas más descriptivas.  
names(poblacion) <- c("provincia", "canton", "parroquia", "total\_nacional", "hombres", "mujeres")

Limpiamos los resultados del censo:

# Filtra el dataframe para eliminar filas con valores NA en la columna 'provincia'  
# y excluye filas donde la columna 'parroquia' contiene las palabras "Total" o "Nacional".  
poblacion <- poblacion %>%   
 filter(!is.na(provincia),   
 str\_detect(parroquia, "^Total|Nacional", negate = TRUE))   
  
# Convierte los valores de las columnas 'provincia', 'canton', y 'parroquia' a mayúsculas para mantener uniformidad.  
poblacion <- poblacion %>%   
 mutate(across(c(provincia, canton, parroquia), str\_to\_upper))

Ahora leemos la codificación DPA del INEC

# Lee la hoja "CODIGOS" del archivo Excel que contiene la codificación DPA.  
dpa <- read\_excel("data/CODIFICACIÓN\_2022.xlsx", sheet = "CODIGOS")

Vamos a acordamos de las funciones join:

# Left Join entre la tabla de población y la tabla de codificación DPA.  
poblacion <- poblacion %>%   
 left\_join(dpa, by = c("provincia" = "DPA\_DESPRO",   
 "canton" = "DPA\_DESCAN",   
 "parroquia" = "DPA\_DESPAR"   
 ))

# Parte 3: Cálculo del Número de Amenidades

Nos queda unir los poligonos a nivel de parroquia por que estan a nivel de zona censal

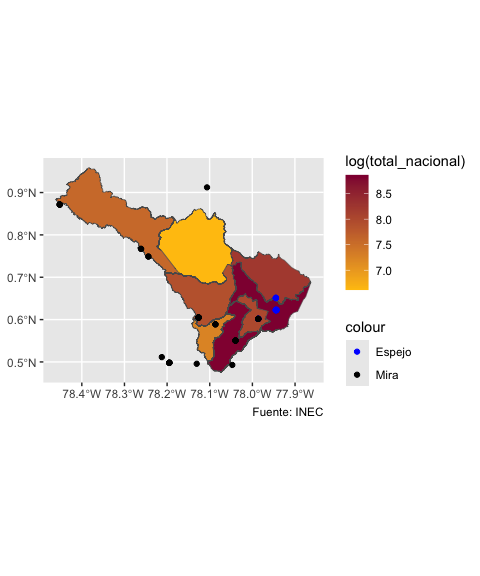
# Crea una nueva columna 'DPA\_PARROQ' con los 6 primeros caracteres de la columna zon  
zonas <- zonas %>%   
 mutate(DPA\_PARROQ = str\_sub(zon, 1, 6))   
  
# Se agrupa las geometrías por 'DPA\_PARROQ' y realiza una unión espacial de las formas para consolidar los polígonos.  
parroquias <- zonas %>%   
 group\_by(DPA\_PARROQ) %>%  
 summarise(Shape = st\_union(Shape))

Unimos los datos de poblacion:

parroquias <- parroquias %>%   
 left\_join(poblacion)

Joining with `by = join\_by(DPA\_PARROQ)`

Mapas de calor



En el mapa de calor, se observa que la mayor concentración de hospitales se encuentra en la ciudad de Tulcán, que es la capital de la provincia de Carchi. Esto es esperado, ya que las capitales provinciales suelen contar con una mayor infraestructura de salud. Además, se destaca la presencia significativa de hospitales en el cantón de Montúfar, que es la segunda parroquia más grande en términos de población del cantón. Esto puede indicar una distribución de servicios de salud que responde a la densidad poblacional y a la importancia relativa de estas localidades dentro de la provincia.

# Parte 4: Cálculo del Ratio de Amenidades por habitante

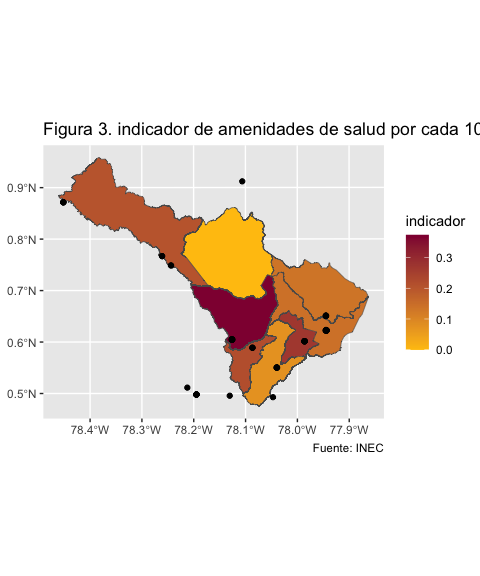
Vamos al indicador de amenidades por cada 100 habitantes:

puntos\_global <- st\_transform(x = puntos\_global, crs = st\_crs(parroquias))  
  
parroquias\_puntos <- st\_join(x = puntos\_global, y = parroquias, join = st\_within)  
  
parroquias\_puntos <- parroquias\_puntos %>%   
 count(DPA\_PARROQ, name = "hospitales") %>%   
 as\_tibble()  
  
parroquias <- parroquias %>%   
 left\_join(parroquias\_puntos)

Joining with `by = join\_by(DPA\_PARROQ)`

parroquias <- parroquias %>%   
 mutate(hospitales = replace\_na(hospitales, 0))

Se calcula el indicador



En los cantones Espejo y Mira se puede observar que el cantón de Mira concentra la mayor cantidad de amenidades de salud. Esto refleja una tendencia común en la cual los centros de salud se localizan en donde existe mayor densidad poblacional.

# Parte 5: Cálculo del indice de Moran para el indicador de Amenidades de salud

vecinos <- poly2nb(parroquias, queen = TRUE)  
  
vecinos\_pesos <- nb2listw(vecinos, style = "W")  
  
matriz <- listw2mat(vecinos\_pesos)  
ncol(matriz)

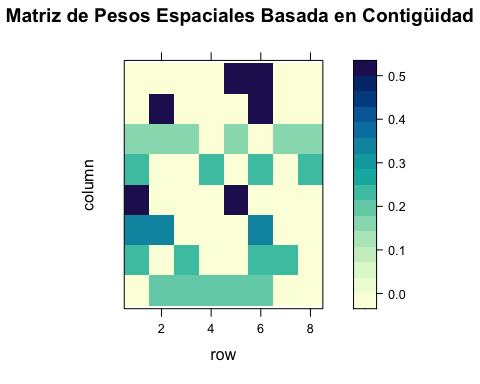
[1] 8

nrow(matriz)

[1] 8

Se visualiza la matriz de vecindad

levelplot(t(matriz[1:8, 1:8]),   
 scales = list(y = list(at = seq(10, 50, by = 10),   
 labels = seq(10, 50, by = 10))),   
 main = "Matriz de Pesos Espaciales Basada en Contigüidad")



Y por último, se calcula el indice de moran

moran <- moran.test(parroquias$hospitales, listw = vecinos\_pesos)   
moran

Moran I test under randomisation  
  
data: parroquias$hospitales   
weights: vecinos\_pesos   
  
Moran I statistic standard deviate = -0.19976, p-value = 0.5792  
alternative hypothesis: greater  
sample estimates:  
Moran I statistic Expectation Variance   
 -0.18170791 -0.14285714 0.03782546

Calculo de indice local de moran

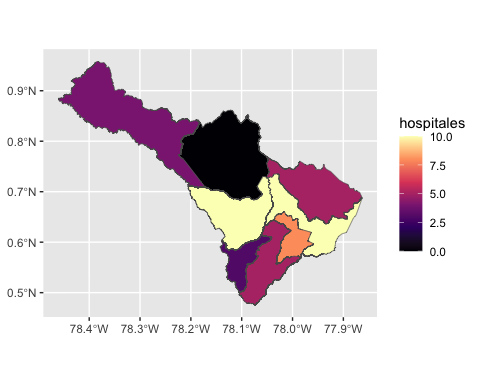
local\_moran <- localmoran(parroquias$hospitales, listw = vecinos\_pesos)

head(local\_moran)

Ii E.Ii Var.Ii Z.Ii Pr(z != E(Ii))  
1 -0.01018923 -0.254730713 0.101249588 0.7685213 0.44217754  
2 -0.85152838 -0.421085465 0.243772496 -0.8718128 0.38331054  
3 -0.06065017 -0.005198586 0.009193886 -0.5783149 0.56305152  
4 0.41484716 -0.075067582 0.231441466 1.0183567 0.30850847  
5 -0.12372635 -0.005198586 0.005171561 -1.6481976 0.09931214  
6 -0.45851528 -0.254730713 0.042187328 -0.9921567 0.32112107

Integramos los resultados del indice de Moran Local y los atributos de objetos resultantes

clusters <- attributes(local\_moran)$quadr  
  
parroquias\_zonas <- bind\_cols(parroquias, clusters)  
  
local\_moran <- local\_moran %>%   
 as\_tibble()   
  
parroquias\_zonas <- parroquias\_zonas %>%   
 bind\_cols(local\_moran)  
  
plot\_1 <- parroquias %>%   
 ggplot() +   
 geom\_sf(aes(fill = hospitales)) +   
 scale\_fill\_viridis\_c(option = "A")   
plot\_1



Se tiene un p-valor mayor que 0.05 por lo que no rechazamos la hipótesis nula de ausencia de autocorrelación espacial. Es decir, existe evidencia estadística de que no existe una autocorrelación espacial significativa en el promedio del número de hospitales por cada 100 habitantes.