Praktik Geocoding dan Big Data

1. Praktik Geocoding dan Big Data untuk analisis fasilitas kesehatan : Evaluasi Pelayanan Kesehatan di Masa Pandemi COVID-19 rancang/buatkan/salin dan tempel *coding* pada sebagai berikut :

install.packages(“sf”)

install.packages(“tidyverse”)

install.packages(“dplyr”)

install.packages(“ggplot2”)

install.packages(“ggspatial”)

library(sf)

library(tidyverse)

library(dplyr)

library(ggplot2)

library(ggspatial)

# 1. Load Data Spasial

rs\_depok <- st\_read("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_RSDepok/RS\_Depok.shp")

adm\_depok <- st\_read("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/Adm\_Depok/SHP\_Depok.shp")

# 2. Pastikan CRS Konsisten

st\_crs(rs\_depok) <- st\_crs(adm\_depok)

# 3. Konversi ke Dataframe untuk Visualisasi

df\_rs <- as.data.frame(st\_coordinates(rs\_depok))

df\_adm <- fortify(adm\_depok)

# 4. Visualisasi Peta RS di Kota Depok

ggplot() +

geom\_sf(data = adm\_depok, fill = "lightgray", color = "black", alpha = 0.5) +

geom\_sf(data = rs\_depok, aes(geometry = geometry), color = "red", size = 2) +

annotation\_scale(location = "br", width\_hint = 0.4) +

annotation\_north\_arrow(location = "tr", which\_north = "true",

pad\_x = unit(0.1, "in"), pad\_y = unit(0.1, "in"),

style = north\_arrow\_fancy\_orienteering) +

theme\_minimal() +

labs(title = "Sebaran Rumah Sakit di Kota Depok",

subtitle = "Data dari SHP RS dan Admin Depok",

caption = "Sumber: Google My Maps")

# 5. Analisis Big Data

names(adm\_depok)

glimpse(adm\_depok)

st\_join(rs\_depok, adm\_depok, left = FALSE)

## 5.1 Spatial Join: Hitung RS per Kecamatan

rs\_per\_kecamatan <- st\_join(rs\_depok, adm\_depok) %>%

group\_by(WADMKC) %>%

summarise(jumlah\_rs = n())

print(rs\_per\_kecamatan)

## 5.2 Buffer Analysis: Radius 2 km dari setiap RS

rs\_buffer <- st\_buffer(rs\_depok, dist = 2000)

ggplot() +

geom\_sf(data = adm\_depok, fill = "lightgray", color = "black", alpha = 0.5) +

geom\_sf(data = rs\_buffer, aes(geometry = geometry), fill = "blue", alpha = 0.3) +

geom\_sf(data = rs\_depok, aes(geometry = geometry), color = "red", size = 2) +

labs(title = "Cakupan Layanan RS di Depok (2 km)") +

theme\_minimal()

## 5.3 Nearest Neighbor Analysis

install.packages("FNN")

library(FNN)

coords\_rs <- st\_coordinates(rs\_depok)

nn\_rs <- get.knn(coords\_rs, k = 1)

rs\_depok$nearest\_dist <- nn\_rs$nn.dist

summary(rs\_depok$nearest\_dist)

## 5.4 Hotspot Analysis

install.packages("spdep")

library(spdep)

coords <- st\_coordinates(rs\_depok)

nb <- knn2nb(knearneigh(coords, k = 5))

listw <- nb2listw(nb, style = "W")

rs\_depok$nearest\_dist <- as.numeric(rs\_depok$nearest\_dist)

local\_moran <- localmoran(rs\_depok$nearest\_dist, listw)

rs\_depok$hotspot <- local\_moran[,1]

install.packages("ggplot2")

library(ggplot2)

ggplot() +

geom\_sf(data = adm\_depok, fill = "lightgray", color = "black", alpha = 0.5) +

geom\_sf(data = rs\_depok, aes(geometry = geometry, color = hotspot), size = 2) +

scale\_color\_viridis\_c() +

labs(title = "Hotspot Analisis Rumah Sakit di Depok") +

theme\_minimal()

# Praktik geocoding dan big data untuk analisis fasilitas pendidikan : Evaluasi Zonasi Penerimaan Peserta Didik Baru

# Load packages

library(sf)

library(ggplot2)

library(dplyr)

library(leaflet)

library(spatstat)

library(spatstat.geom) # untuk as.owin.sf()

library(terra)

# Load shapefiles

sekolah\_sf <- st\_read("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_SekolahNegeri\_Depok/SekolahNegeri\_Depok/SekolahNegeri\_Depok.shp")

batas\_sf <- st\_read("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/Adm\_Depok/SHP\_Depok.shp")

# Hilangkan Z dimension jika ada

sekolah\_sf <- st\_zm(sekolah\_sf)

batas\_sf <- st\_zm(batas\_sf)

# Transform to UTM Zone 48S (EPSG:32748)

batas\_proj <- st\_transform(batas\_sf, 32748)

sekolah\_proj <- st\_transform(sekolah\_sf, 32748)

# Convert batas to owin langsung dari sf

win <- as.owin(batas\_proj)

# Kernel Density Estimation

density\_sekolah <- density.ppp(ppp\_sekolah, sigma = 500, eps = c(100, 100))

# Convert density raster to dataframe

density\_raster <- rast(density\_sekolah)

density\_df <- as.data.frame(density\_raster, xy = TRUE)

names(density\_df) <- c("x", "y", "value")

# Visualize with ggplot

peta <- ggplot() +

geom\_sf(data = batas\_proj, fill = NA, color = "black", linetype = "dashed") +

geom\_tile(data = density\_df, aes(x = x, y = y, fill = value), alpha = 0.7) +

scale\_fill\_viridis\_c(labels = scales::label\_number(scale\_cut = scales::cut\_short\_scale())) +

geom\_sf(data = sekolah\_proj, color = "red", size = 3) +

theme\_minimal() +

labs(title = "Peta Sekolah Negeri di Kota Depok", subtitle = "Analisis Kepadatan Sekolah (KDE)")

print(peta)

# Create interactive map

leaflet() %>%

addTiles() %>%

addPolygons(data = batas\_sf, color = "black", weight = 2, fill = FALSE) %>%

addCircleMarkers(data = sekolah\_sf, color = "red", radius = 10, label = sekolah\_sf$Nama)

# Praktik Geocoding dan Big Data untuk analisis kependudukan : Tingkat Bahaya Pernikahan Dini

# Instalasi Koding

install.packages(c("sf", "raster", "terra", "tidyverse", "exactextractr"))

library(raster)

library(sf)

library(terra)

library(tidyverse)

library(exactextractr)

# Memuat file .shp

depok\_shp <- st\_read("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_PendudukDepok/Penduduk\_Depok.shp")

# Data Penduduk

births\_raster <- raster("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_PendudukDepok/Raster\_Penduduk/Indonesia\_1km\_Births/IDN\_births\_pp\_v2\_2015.tif")

pregs\_raster <- raster("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_PendudukDepok/Raster\_Penduduk/Indonesia\_1km\_Pregnancies/IDN\_pregs\_pp\_v2\_2015.tif")

agesex\_raster <- raster("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_PendudukDepok/Raster\_Penduduk/idn\_pd\_2020\_1km\_UNadj.tif") # Optional jika breakdown lebih detail

# Ubah shapefile jadi 2D

depok\_shp <- st\_zm(depok\_shp, drop = TRUE, what = "ZM")

# Samakan CRS agar cocok

depok\_shp <- st\_transform(depok\_shp, crs(pregs\_raster))

# Pregnancies (kehamilan)

pregs\_depok <- terra::mask(terra::crop(pregs\_raster, depok\_shp), depok\_shp)

# Births (kelahiran)

births\_depok <- terra::mask(terra::crop(births\_raster, depok\_shp), depok\_shp)

# Age and Sex Structure

agesex\_depok <- terra::mask(terra::crop(agesex\_raster, depok\_shp), depok\_shp)

#Score and Note

total\_births <- cellStats(births\_depok, sum, na.rm = TRUE)

total\_pregs <- cellStats(pregs\_depok, sum, na.rm = TRUE)

total\_agesex <- cellStats(agesex\_depok,sum, na.rm = TRUE)

cat("Total kelahiran di Depok:", round(total\_births), "\n")

cat("Total kehamilan di Depok:", round(total\_pregs), "\n")

cat("Total Umur dan jenis kelamin di Depok:", round(total\_agesex), "\n")

#Visualize Plot

plot(births\_depok, main = "Distribusi Kelahiran di Kota Depok")

plot(pregs\_depok, main = "Distribusi Kehamilan di Kota Depok")

plot(agesex\_depok, main = "Distribusi Umur dan Jenis Kelamin di Kota Depok")

#Visualize Plot

plot(births\_depok, main = "Distribusi Kelahiran di Kota Depok")

plot(pregs\_depok, main = "Distribusi Kehamilan di Kota Depok")

plot(agesex\_depok, main = "Distribusi Umur dan Jenis Kelamin di Kota Depok")

# wanita usia 15–19

agesex\_stack <- stack("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_PendudukDepok/Raster\_Penduduk/idn\_f\_15\_49\_2015\_1km.tif")

nlayers(agesex\_stack)

# Lihat info band

agesex\_stack

# Misal band 10 = wanita usia 15–19

female\_15\_19 <- agesex\_stack[[1]]

female\_15\_19\_depok <- mask(crop(female\_15\_19, depok\_shp), depok\_shp)

# Visualisasi

plot(female\_15\_19\_depok, main = "Perempuan Usia 15–19 di Depok")

# Ubah raster menjadi data.frame

births\_df <- as.data.frame(rasterToPoints(births\_depok))

colnames(births\_df) <- c("lon", "lat", "births")

pregs\_df <- as.data.frame(rasterToPoints(pregs\_depok))

colnames(pregs\_df) <- c("lon", "lat", "pregnancies")

teen\_df <- as.data.frame(rasterToPoints(female\_15\_19\_depok))

colnames(teen\_df) <- c("lon", "lat", "female\_15\_19")

# Gabungkan ketiganya

merged\_df <- reduce(list(births\_df, pregs\_df, teen\_df), full\_join, by = c("lon", "lat"))

head(merged\_df)

#KONVERSI KE SPATIAL SF

coordinates(merged\_df) <- ~lon + lat

proj4string(merged\_df) <- CRS("+proj=longlat +datum=WGS84")

merged\_sf <- st\_as\_sf(merged\_df)

# SPATIAL WEIGHTS (1 km radius)

install.packages("spdep", dependencies = TRUE)

library(spdep)

coords <- st\_coordinates(merged\_sf)

nb <- dnearneigh(coords, d1 = 0, d2 = 1000)

lw <- nb2listw(nb, style = "W", zero.policy = TRUE)

# LOCAL MORAN UNTUK JUMLAH KEHAMILAN

library(sf)

library(spdep)

library(dplyr)

hitung\_local\_moran <- function(sf\_data, value\_col, d1 = 0, d2 = 1000) {

# Pastikan bentuknya sf

if (!inherits(sf\_data, "sf")) {

sf\_data <- st\_as\_sf(sf\_data)

}

# Ubah ke numeric (jaga-jaga kalau masih list/character)

sf\_data[[value\_col]] <- as.numeric(sf\_data[[value\_col]])

# Hapus NA

sf\_clean <- sf\_data %>% filter(!is.na(.data[[value\_col]]))

# Ambil koordinat

coords <- st\_coordinates(sf\_clean)

# Buat spatial weights

nb <- dnearneigh(coords, d1 = d1, d2 = d2)

lw <- nb2listw(nb, style = "W", zero.policy = TRUE)

# Hitung Local Moran

local\_moran <- localmoran(sf\_clean[[value\_col]], lw, zero.policy = TRUE)

# Tambahkan hasil ke sf

sf\_clean$Ii <- local\_moran[, 1]

sf\_clean$Z\_Ii <- local\_moran[, 4]

sf\_clean$hotspot <- "Not Significant"

sf\_clean$hotspot[sf\_clean$Z\_Ii > 1.96] <- "Hotspot"

sf\_clean$hotspot[sf\_clean$Z\_Ii < -1.96] <- "Coldspot"

return(sf\_clean)

}

# Misal merged\_sf adalah hasil gabungan shapefile dan data WorldPop

result\_sf <- hitung\_local\_moran(merged\_sf, "pregnancies")

# Plot hasilnya

library(ggplot2)

ggplot(result\_sf) +

geom\_sf(aes(color = hotspot), size = 6) +

scale\_color\_manual(values = c("Hotspot" = "red", "Coldspot" = "blue", "Not Significant" = "grey")) +

labs(title = "Hotspot Potensi Pernikahan Dini di Depok", color = "Kategori") +

theme\_minimal()

1. Praktik Geocoding dan Big Data Untuk Analisis Sosial dan Politik : Partisipasi Pemilihan Gubernur Jawa Barat 2024

# Instal dan panggil library yang dibutuhkan

install.packages(c("sf", "raster", "terra", "tidyverse", "ggplot2", "tmap", "exactextractr"))

library(sf)

library(raster)

library(terra)

library(tidyverse)

library(tmap)

library(exactextractr)

# 1. SHP Penduduk Laki-laki dan Perempuan

penduduk <- st\_read("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_Sospol\_JawaBarat/Penduduk\_Jabar/Penduduk\_Jabar.shp")

# 2. Raster Kepadatan Penduduk

kepadatan <- raster("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_Sospol\_JawaBarat/idn\_pd\_2020\_1km.tif")

# 3. CSV Partisipasi Pemilih

partisipasi <- read.csv("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_Sospol\_JawaBarat/Partisipasi\_Pilkada\_Jabar.csv")

# Baca ulang CSV

partisipasi\_raw <- read.csv("E:/NGI\_Mentor\_Data/Praktikum\_Geocoding\_BigData/SHP\_Sospol\_JawaBarat/Partisipasi\_Pilkada\_Jabar.csv", header = TRUE)

# Pisahkan kolom jadi 4 bagian

partisipasi <- partisipasi\_raw %>%

separate(Kab.Kota.Persentase.Longitude.Latitude, into = c("KabKota", "Persentase", "Longitude", "Latitude"), sep = ";") %>%

mutate(

Persentase = as.numeric(Persentase),

Longitude = as.numeric(Longitude),

Latitude = as.numeric(Latitude)

)

# Gabungkan berdasarkan nama kabupaten/kota

penduduk\_join <- penduduk %>%

left\_join(partisipasi, by = c("NAME\_2" = "KabKota"))

library(sf)

partisipasi\_sf <- st\_as\_sf(partisipasi, coords = c("Longitude", "Latitude"), crs = 4326)

# Plot titik geocoding dan boundary

tmap\_mode("view")

tm\_shape(penduduk) + tm\_borders() +

tm\_shape(partisipasi\_sf) + tm\_dots(col = "Persentase", palette = "viridis", size = 1)

tm\_shape(penduduk\_join) +

tm\_polygons("Persentase", palette = "Blues", title = "Partisipasi Pemilih (%)") +

tm\_layout(title = "Peta Partisipasi Pilgub Jabar 2024")

library(exactextractr)

# Buat kolom 'kepadatan\_rata' dari raster ke polygon

penduduk\_join$kepadatan\_rata <- exact\_extract(kepadatan, penduduk\_join, 'mean')

ggplot(penduduk\_join, aes(x = kepadatan\_rata, y = Persentase)) +

geom\_point() +

geom\_smooth(method = "lm") +

labs(x = "Rata-rata Kepadatan", y = "Partisipasi (%)",

title = "Korelasi Kepadatan Penduduk vs Partisipasi")

library(cluster)

cluster\_data <- na.omit(penduduk\_join[, c("Persentase", "kepadatan\_rata")])

cluster\_data <- penduduk\_join %>%

st\_drop\_geometry() %>% # Hilangkan geometri agar kmeans tidak bingung

select(Persentase, kepadatan\_rata) %>% # Pilih hanya kolom numerik

drop\_na() # Hapus baris yang mengandung NA

str(cluster\_data)

penduduk\_join <- penduduk\_join %>%

mutate(row\_id = row\_number())

cluster\_data <- penduduk\_join %>%

st\_drop\_geometry() %>%

select(row\_id, Persentase, kepadatan\_rata) %>%

drop\_na()

kmeans\_result <- kmeans(scale(cluster\_data[, c("Persentase", "kepadatan\_rata")]), centers = 3)

cluster\_data$cluster <- kmeans\_result$cluster

penduduk\_join <- penduduk\_join %>%

left\_join(cluster\_data %>% select(row\_id, cluster), by = "row\_id") %>%

mutate(cluster = as.factor(cluster))

tm\_shape(penduduk\_join) +

tm\_polygons("cluster", palette = "Set2", title = "Cluster K-Means") +

tm\_layout(title = "Clustering Wilayah berdasarkan Partisipasi & Kepadatan")

library(spdep)

library(sf)

library(sp)

library(spdep)

# 1. Perbaiki geometri

penduduk\_valid <- st\_make\_valid(penduduk\_join)

# 2. Ubah ke Spatial

penduduk\_sp <- as(penduduk\_valid, "Spatial")

# 3. Buat neighbors

nb <- poly2nb(penduduk\_sp)

lw <- nb2listw(nb, style = "W")

# Indeks baris tanpa NA

id\_valid <- which(!is.na(penduduk\_join$Persentase))

# Subset data dan buat neighbors

penduduk\_sp <- as(penduduk\_join[id\_valid, ], "Spatial")

nb <- poly2nb(penduduk\_sp)

library(sf)

library(sp)

library(spdep)

# 1. Hapus baris NA untuk partisipasi

penduduk\_no\_na <- penduduk\_join %>%

filter(!is.na(Persentase))

# 2. Perbaiki geometri secara valid

penduduk\_no\_na <- st\_make\_valid(penduduk\_no\_na)

# 3. Hapus poligon yang tetap tidak valid (meskipun sudah diperbaiki)

penduduk\_no\_na <- penduduk\_no\_na[st\_is\_valid(penduduk\_no\_na), ]

# 4. Konversi ke Spatial (wajib untuk poly2nb)

penduduk\_sp <- as(penduduk\_no\_na, "Spatial")

# 5. Bangun neighbours list

nb <- poly2nb(penduduk\_sp)

# 6. Buat spatial weights

lw <- nb2listw(nb, style = "W")

# 7. Uji Moran’s I

moran.test(penduduk\_sp$Persentase, lw)

plot(st\_geometry(penduduk\_no\_na), col = "lightblue", main = "Poligon Valid untuk Moran's I")

moran.plot(penduduk\_sp$Persentase, lw,

labels = row.names(penduduk\_sp),

xlab = "Partisipasi",

ylab = "Lag Partisipasi",

main = "Moran Scatterplot Partisipasi Pemilih")

local\_moran <- localmoran(penduduk\_sp$Persentase, lw)

penduduk\_no\_na$local\_I <- local\_moran[,1]

penduduk\_no\_na$p\_value <- local\_moran[,5]

# Kategorisasi hotspot / coldspot

penduduk\_no\_na$cluster\_type <- NA

penduduk\_no\_na$cluster\_type[penduduk\_no\_na$local\_I > 0 & penduduk\_no\_na$p\_value < 0.05] <- "Hotspot"

penduduk\_no\_na$cluster\_type[penduduk\_no\_na$local\_I < 0 & penduduk\_no\_na$p\_value < 0.05] <- "Coldspot"

penduduk\_no\_na$cluster\_type[is.na(penduduk\_no\_na$cluster\_type)] <- "Non-signifikan"

# Plot pakai tmap

library(tmap)

tmap\_mode("view")

tm\_shape(penduduk\_no\_na) +

tm\_polygons("cluster\_type", palette = c("red", "blue", "gray"),

title = "LISA Cluster Map") +

tm\_layout(title = "Hotspot dan Coldspot Partisipasi Pilgub Jabar 2024")

# 6. Buat spatial weights

lw <- nb2listw(nb, style = "W")

# 7. Uji Moran’s I

moran.test(penduduk\_sp$Persentase, lw)

plot(st\_geometry(penduduk\_no\_na), col = "lightblue", main = "Poligon Valid untuk Moran's I")

moran.plot(penduduk\_sp$Persentase, lw,

labels = row.names(penduduk\_sp),

xlab = "Partisipasi",

ylab = "Lag Partisipasi",

main = "Moran Scatterplot Partisipasi Pemilih")

local\_moran <- localmoran(penduduk\_sp$Persentase, lw)

penduduk\_no\_na$local\_I <- local\_moran[,1]

penduduk\_no\_na$p\_value <- local\_moran[,5]

# Kategorisasi hotspot / coldspot

penduduk\_no\_na$cluster\_type <- NA

penduduk\_no\_na$cluster\_type[penduduk\_no\_na$local\_I > 0 & penduduk\_no\_na$p\_value < 0.05] <- "Hotspot"

penduduk\_no\_na$cluster\_type[penduduk\_no\_na$local\_I < 0 & penduduk\_no\_na$p\_value < 0.05] <- "Coldspot"

penduduk\_no\_na$cluster\_type[is.na(penduduk\_no\_na$cluster\_type)] <- "Non-signifikan"

# Plot pakai tmap

library(tmap)

tmap\_mode("view")

tm\_shape(penduduk\_no\_na) +

tm\_polygons("cluster\_type", palette = c("red", "blue", "gray"),

title = "LISA Cluster Map") +

tm\_layout(title = "Hotspot dan Coldspot Partisipasi Pilgub Jabar 2024")