Analisis Spasial dengan R

10 Oktober 2022

Septian Rahardiantoro

Outline

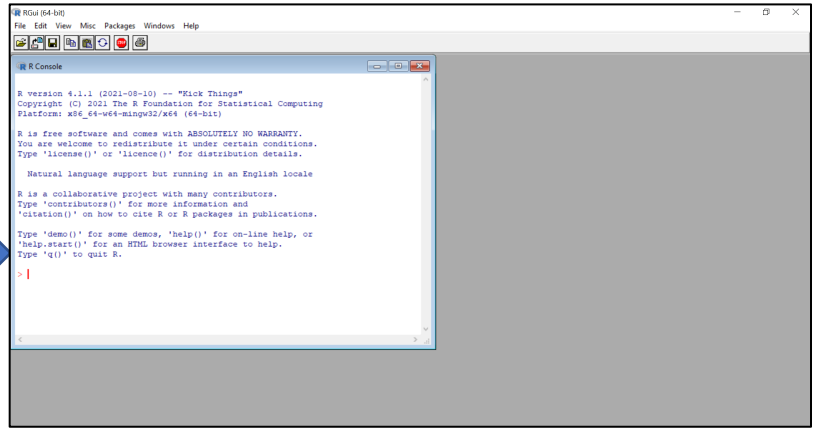
- Pengenalan R
- Plot Objek Spasial
- Autokorelasi Spasial (Spatial Autocorrelation)

Pengenalan R

- R adalah implementasi sebuah lingkungan komputasi dan pemrograman bahasa statistika dengan menggunakan Bahasa S
- Salah satu software statistika yang multifungsi:
  - Analisis Statistika Dasar
  - Analisis Statistika Lanjut
  - Pemrograman
  - Data mining
  - Kemampuan pembuatan grafik yang canggih
  - dsb
- R bersifat open source (tidak berbayar dan tidak perlu lisensi) dengan package-package analisis yang selalu up-to-date

Menjalankan R

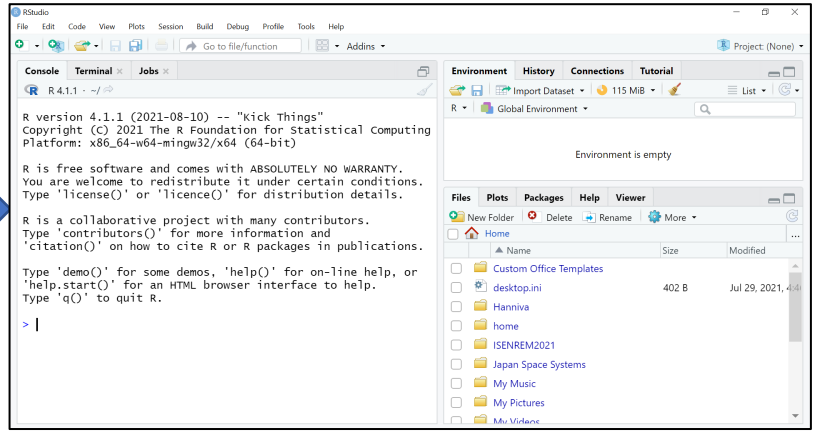
https://www.r-project.org



R-GUI (R-Graphical User Interface)

- Merupakan software R utama yang dapat digunakan. Di R-GUI, Anda dapat menulis program dan menjalankan kode secara independen dari program komputer lainnya.
- Lebih ringan dalam menjalankan program
- Kekurangannya tidak ada fasilitas autocorrect dan code suggestion pada script atau console

https://www.rstudio.com



rstudio

- Merupakan software tambahan untuk menjalankan program R. rstudio perlu digunakan bersama R
- rstudio memungkinkan pengguna untuk mengembangkan dan mengedit program di R dengan mendukung sejumlah besar paket statistik, grafik berkualitas lebih tinggi, dan kemampuan untuk mengelola ruang kerja Anda.
- Software rstudio lebih berat digunakan (lebih banyak memori yang digunakan)

R dan RStudio bukan versi terpisah dari program yang sama, dan tidak dapat saling menggantikan. R dapat digunakan tanpa RStudio, tetapi RStudio tidak dapat digunakan tanpa R.

Plot Objek Spasial

- Siapkan package yang diperlukan
  - rgdal
  - spdep
  - raster

- Package → suatu modul program yang digunakan spesifik untuk mengolah data dengan metode tertentu
- Package yang belum ada harus diinstall pada R, sebelum dapat digunakan
- Setelah diinstall, maka perlu juga package tersebut diaktifkan untuk digunakan

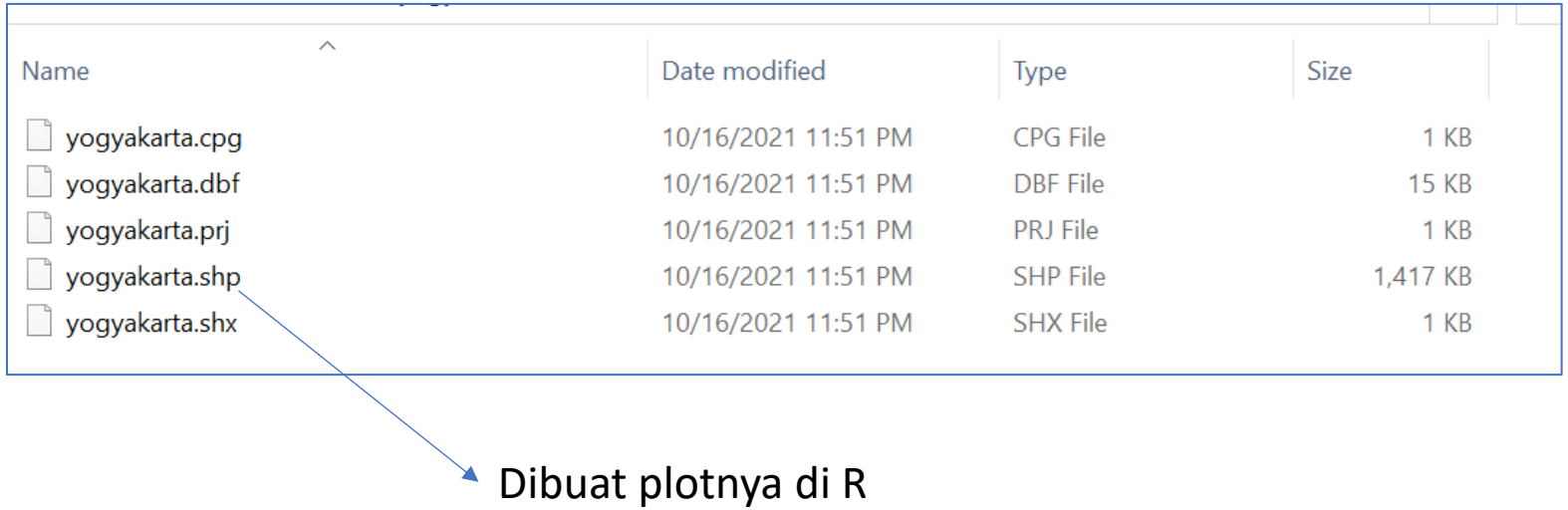
- Cara menginstall package

- Melalui sintax di console (harus tersambung internet)

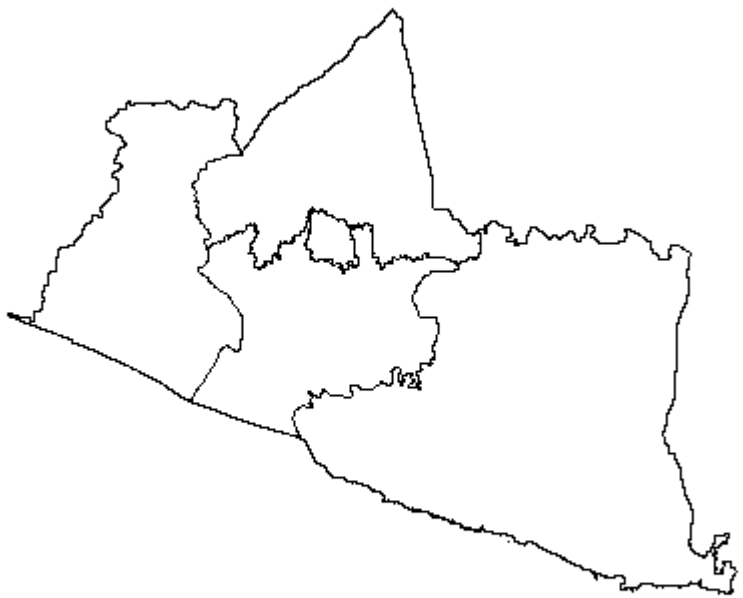
```
install.packages("<nama packages>")
```

Aplikasi R dalam Plot Data Spasial

- Gunakan file yogyakarta.shp
- Akan dibuat plot Kab/Kota di Provinsi Yogyakarta



```
#open shp file
library(rgdal)
yogya <- readOGR(dsn='/Users/gad/Documents/SRO/FKM UI/yogyakarta',layer='yogyakarta')

View(yogya@data) #melihat data yang terkandung dalam shp file
str(yogya@data) #melihat struktur data shp file

plot(yogya) #plot yogya
```



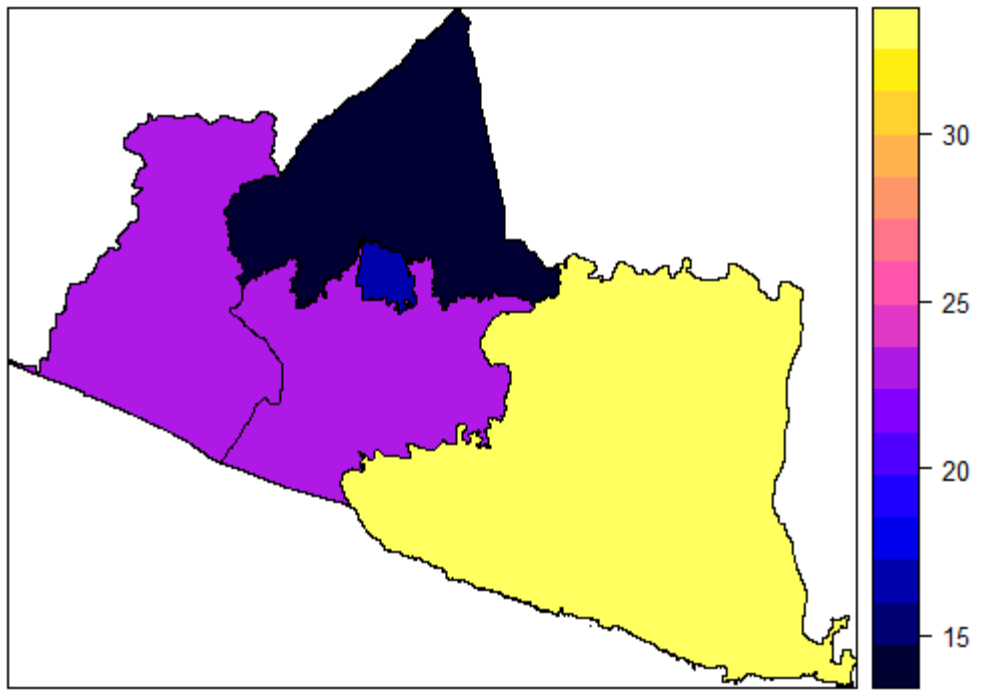
Plot Spasial berdasarkan nilai suatu peubah

```
> str(yogya@data) #melihat struktur data shp file
'data.frame':   5 obs. of  16 variables:
 $ NAMOBJ    : chr  "Bantul" "Gunungkidul" "Kota Yogyakarta" "Kulon Progo" ...
 $ X         : chr  "110Â° 21' 38.314\" E" "110Â° 35' 53.844\" E" "110Â° 22' 28$
 $ Y         : chr  "7Â° 53' 58.604\" S" "7Â° 59' 37.394\" S" "7Â° 48' 11.822\"$
 $ provcode  : chr  "14" "14" "14" "14" ...
 $ provinsi  : chr  "DI Yogyakarta" "DI Yogyakarta" "DI Yogyakarta" "DI Yogyaka$
 $ NAMOBJ_1  : chr  "Bantul" "Gunungkidul" "Kota Yogyakarta" "Kulon Progo" ...
 $ stunting  : chr  "22.89%" "32.51%" "16.93%" "22.65%" ...
 $ DiareDGU5 : chr  "13.56%" "11.58%" "9.34%" "7.63%" ...
 $ ISPADG_U5 : chr  "9.85%" "7.25%" "15.26%" "5.74%" ...
 $ BAB_jamban: chr  "97.52%" "93.82%" "98.99%" "96.30%" ...
 $ CT_Benar  : chr  "46.19%" "47.33%" "53.16%" "40.45%" ...
 $ K4        : chr  "91.24%" "86.32%" "92.82%" "91.28%" ...
 $ imunisasi : chr  "68.67%" "96.05%" "96.90%" "84.04%" ...
 $ topografi : chr  "3" "3" "3" "3" ...
 $ miskin    : num  13.43 17.12 6.98 18.3 7.65
 $ KEK_Hamil : chr  "19.83%" "16.89%" "17.32%" "42.90%" ...
```

Akan dibuat plot spasial berdasarkan nilai peubah stunting

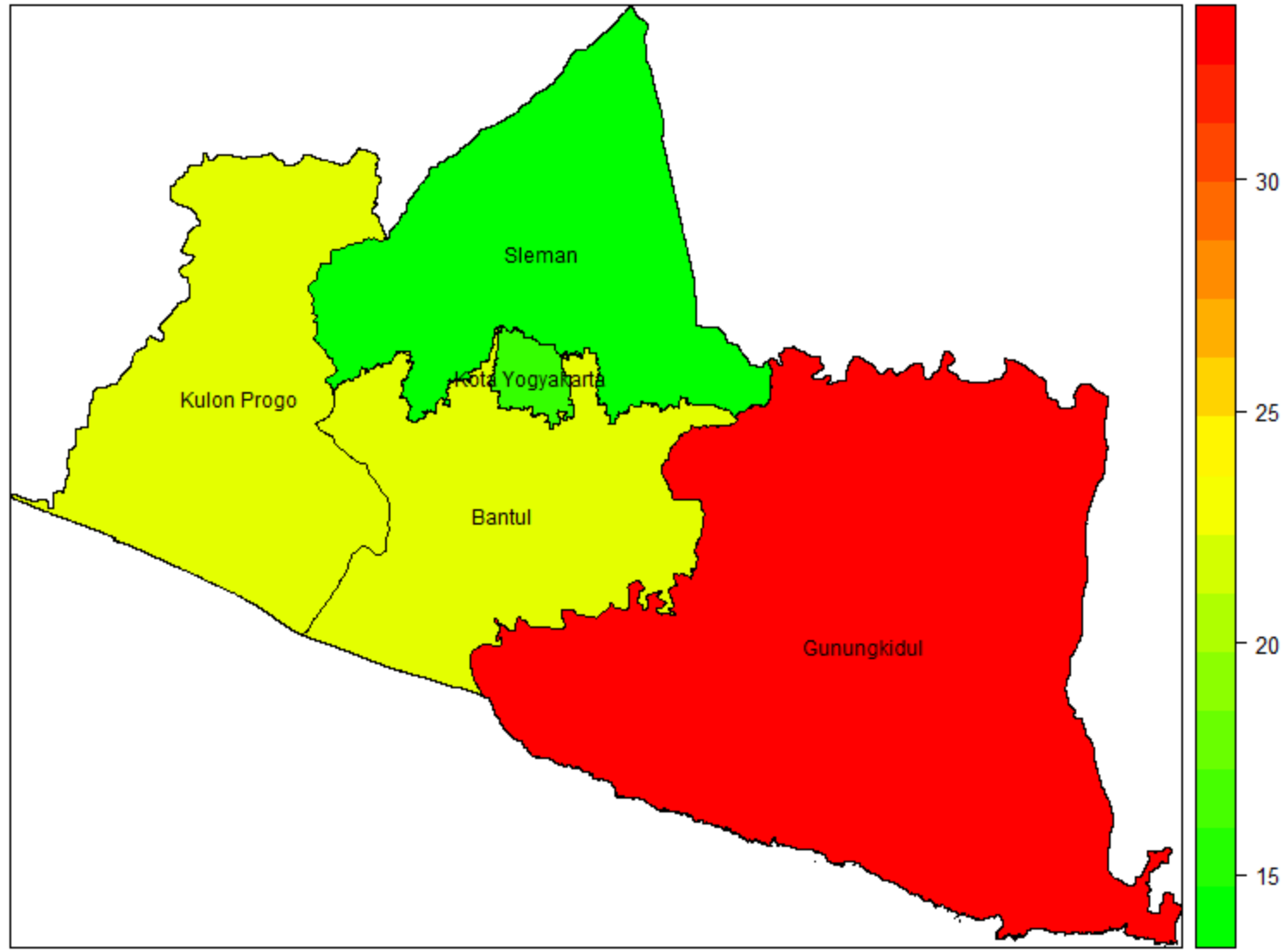
Namun perhatikan class nya bukan numerik

```
#penyesuaian peubah stunting ke bentuk numerik
yogya@data$stunting <- as.numeric(abbreviate(yogya@data$stunting,5,strict=T))

#spatial plot
spplot(yogya,"stunting")
```



Penyesuaian warna plot dan penambahan label

```
k <- coordinates(yogya) #koordinat pusat kab/kota

#pendefinisian label
sp.label <- function(x, label) {list("sp.text", k, label,cex = 0.75)}
NUMB.sp.label <- function(x) {sp.label(x, as.vector(x@data$NAMOBJ))}
make.NUMB.sp.label <- function(x) {do.call("list", NUMB.sp.label(x))}

colfunc1 <- colorRampPalette(c("green","yellow","red"))
kk <- 30
color2<-colfunc1(kk)
spplot(yogya,"stunting",col.regions=color2,sp.layout=make.NUMB.sp.label(yogya))
```


Sleman
Kota Yogyakarta
Kulon Progo
Bantul
Gunungkidul
30
25
20
15

Autokorelasi Spasial

- Ketergantungan antar observasi yang disebabkan adanya **efek lokasi** (spasial)

- Dapat dicek dengan menelusuri keterkaitan antar observasi berdasarkan kedekatan lokasinya

- Tahapan pengecekan autokorelasi spasial (dengan Indeks Moran)
  - Pendefinisian kedekatan lokasi
  - Pembentukan matriks pembobot
  - Indeks Moran

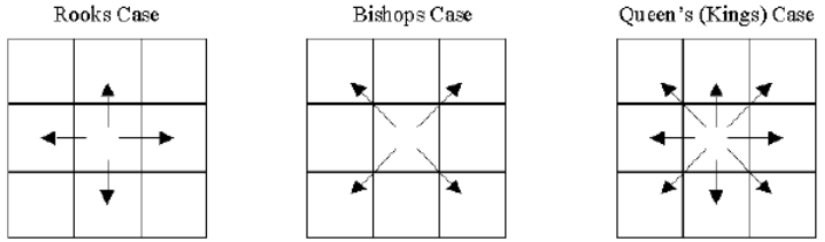
- Tahapan
  1. Pendefinisian kedekatan lokasi (*Choose a neighborhood criterion*)
     – Which areas are linked?
  2. Pembentukan matriks pembobot (*Assign weights to the areas that are linked*)
     – Create a spatial weights matrix
  3. Run statistical test to examine spatial autocorrelation → Index Moran

Tahapan 1

Neighborhood criterion:

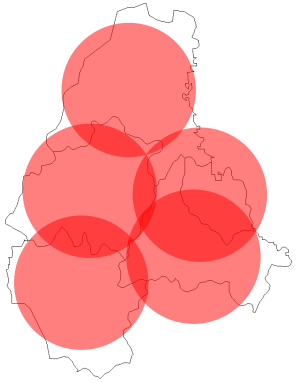
Neighborhoods can be defined in a number of ways

1. Contiguity (common boundary)

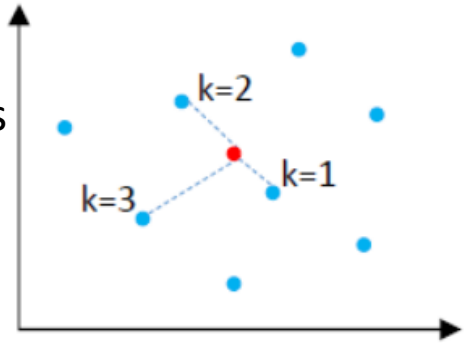


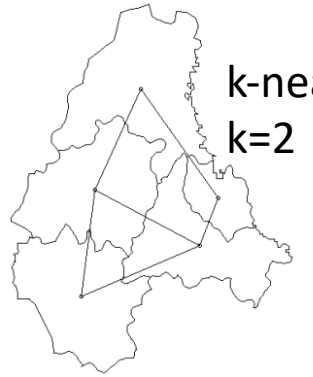
2. Distance (distance band, k-nearest neighbors)

distance band



k-nearest neighbors





k-nearest neighbors
k=2

3. General weight (social distance, distance decay)

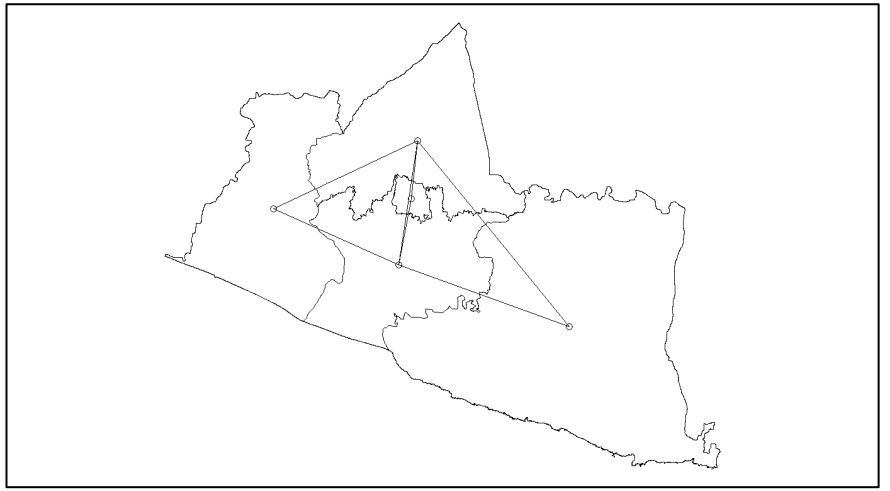
Aplikasi di R

1. Contiguity (common boundary)

QUEEN

```
###queen contiguity
library(spdep)
w <- poly2nb(yogya)
w

k <- coordinates(yogya)
plot(yogya)
plot(w,k,add=T)
```

```
> w
Neighbour list object:
Number of regions: 5
Number of nonzero links: 14
Percentage nonzero weights: 56
Average number of links: 2.8
```

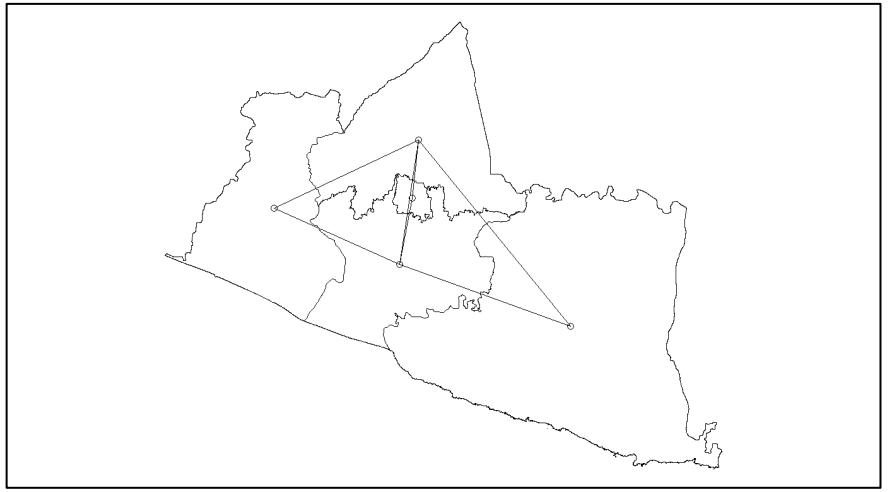


Queen contiguity

ROOK

```
###rook contiguity
w2 <- poly2nb(yogya,queen=F)
w2

plot(yogya)
plot(w2,k,add=T)
```

```
> w2
Neighbour list object:
Number of regions: 5
Number of nonzero links: 14
Percentage nonzero weights: 56
Average number of links: 2.8
```

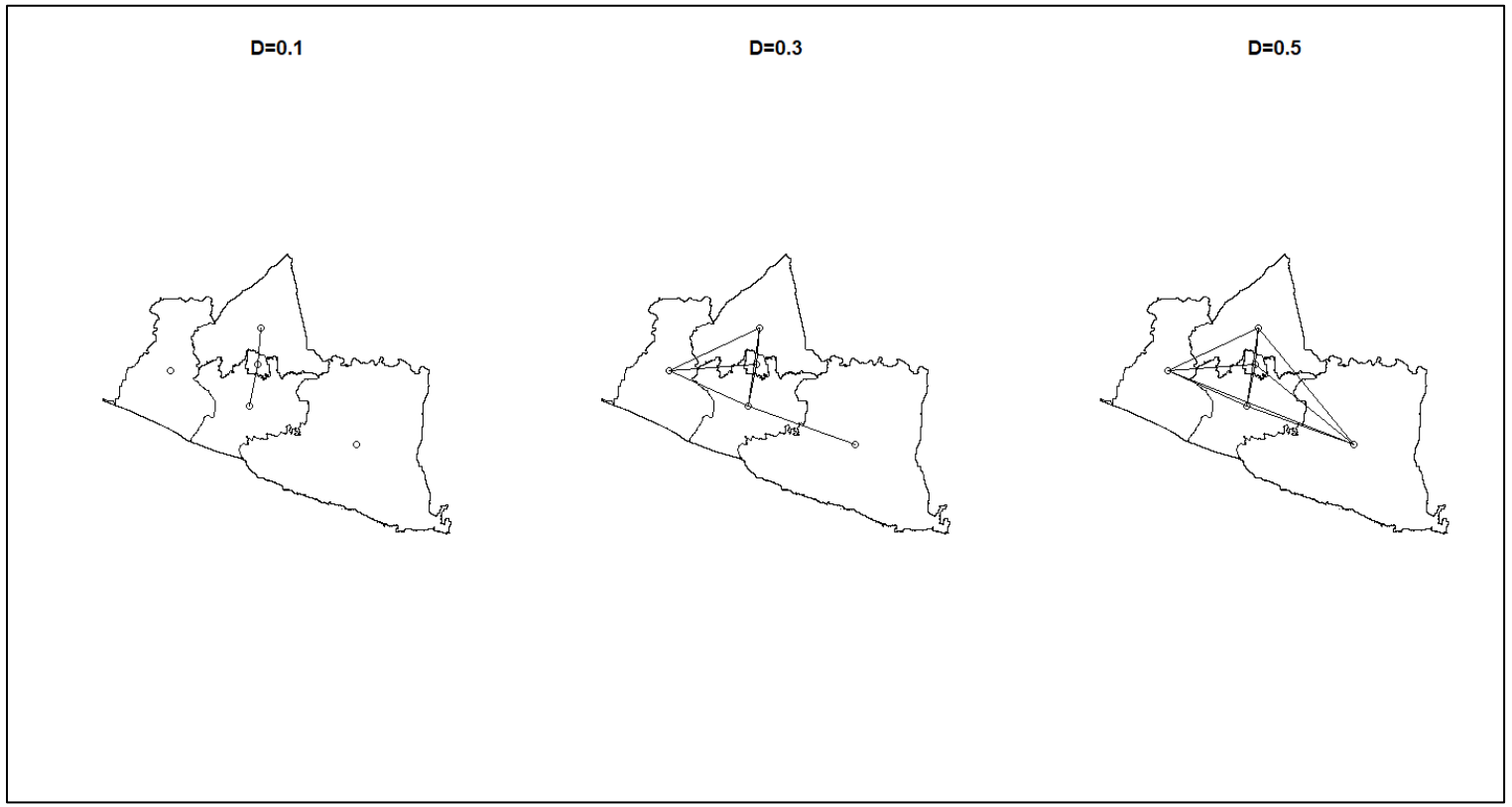


Rook contiguity

2. Distance (distance band, k-nearest neighbors)

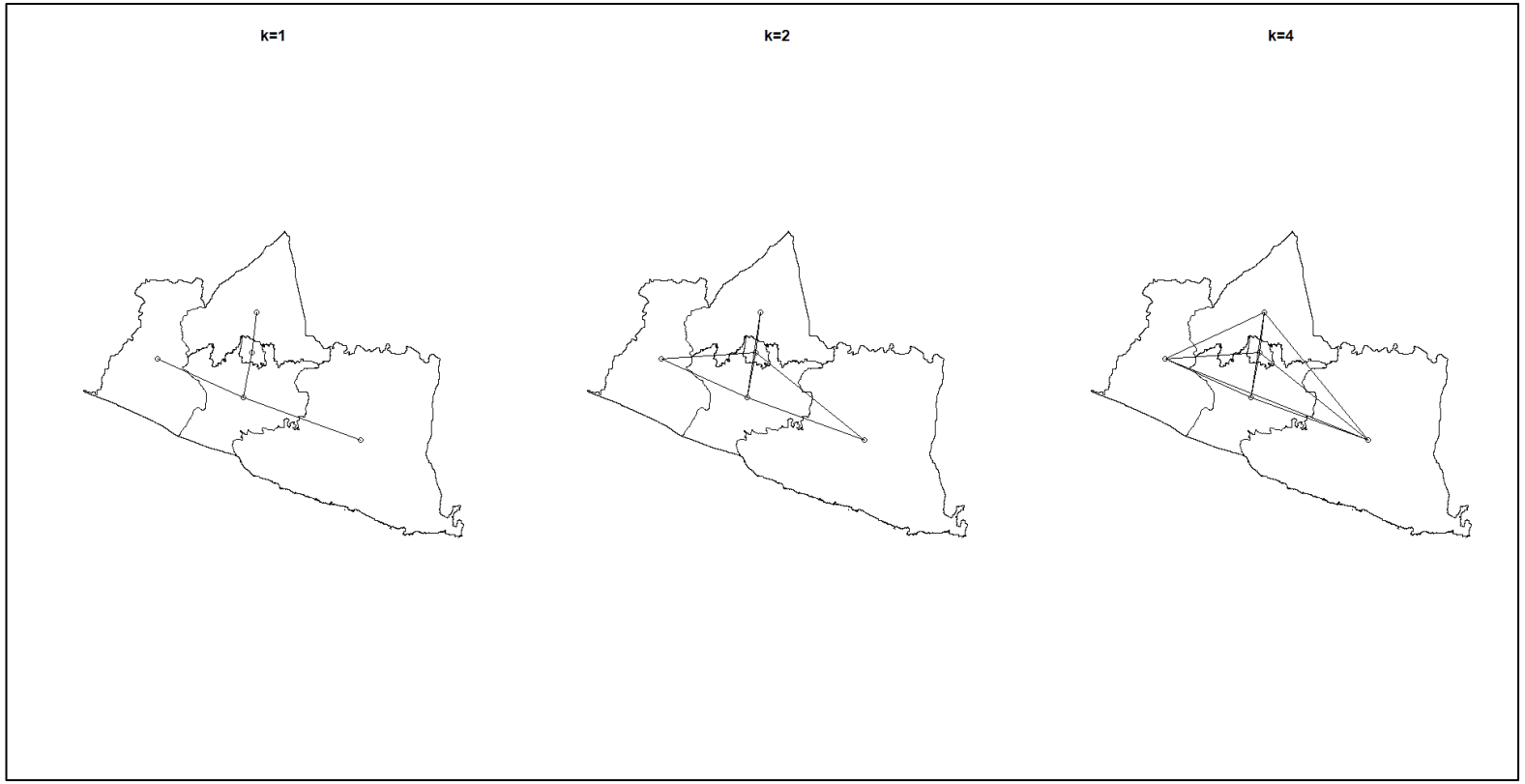
Distance Band

```
###distance band
coords<-coordinates(yogya)

nb1 <- dnearneigh(coords, 0, 0.1) #contiguity by distance = 0.1
nb2 <- dnearneigh(coords, 0, 0.3) #contiguity by distance = 0.3
nb3 <- dnearneigh(coords, 0, 0.5) #contiguity by distance = 0.5

par(mfrow=c(1,3))
plot(yogya,main="D=1")
plot(nb1, coords, add=T)
plot(yogya,main="D=2")
plot(nb2, coords, add=T)
plot(yogya,main="D=4")
plot(nb3, coords, add=T)
```



k-Nearest Neighbors

```
###k-nn
coords<-coordinates(yogya)
kn1<-knn2nb(knearneigh(coords, k=1))
kn2<-knn2nb(knearneigh(coords, k=2))
kn4<-knn2nb(knearneigh(coords, k=4))
par(mfrow=c(1,3))
plot(yogya,main="k=1")
plot(kn1, coords, add=T)
plot(yogya,main="k=2")
plot(kn2, coords, add=T)
plot(yogya,main="k=4")
plot(kn4, coords, add=T)
```



Tahapan 2

Pembentukan Matriks Pembobot

- Setelah mendefinisikan konsep ketetanggaan, maka selanjutnya dapat dibentuk matriks pembobot spasial

- Matriks pembobot spasial secara umum memiliki dua bentuk
  - Binary weights matrix
    - Antar dua objek yang bertetangga dinyatakan dengan nilai 1, dan nilai 0 untuk sepasang objek yang tidak bertetangga
  - Row-standardized weights matrix
    - Digunakan untuk membuat bobot proporsional dalam kasus di mana objek memiliki jumlah tetangga yang tidak sama

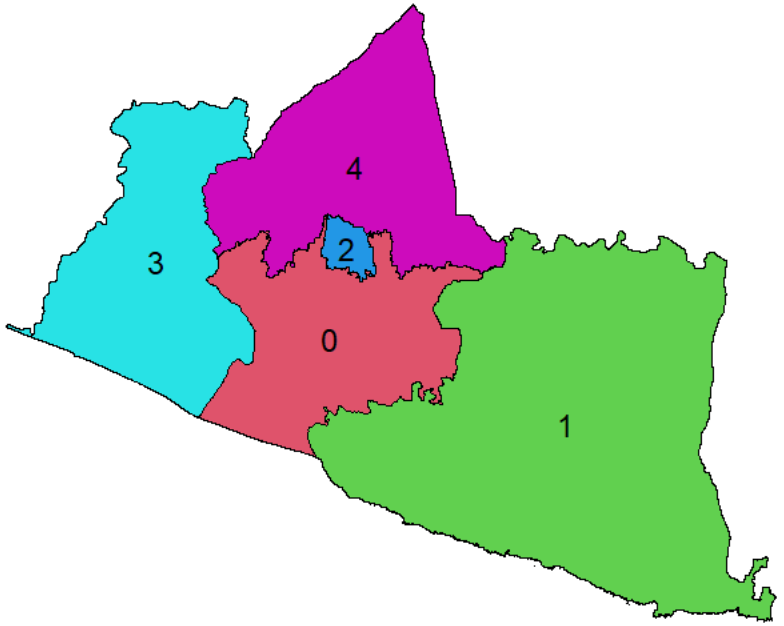
- Memungkinkan terdapat objek yang tidak memiliki tetangga → perlu perlakuan khusus

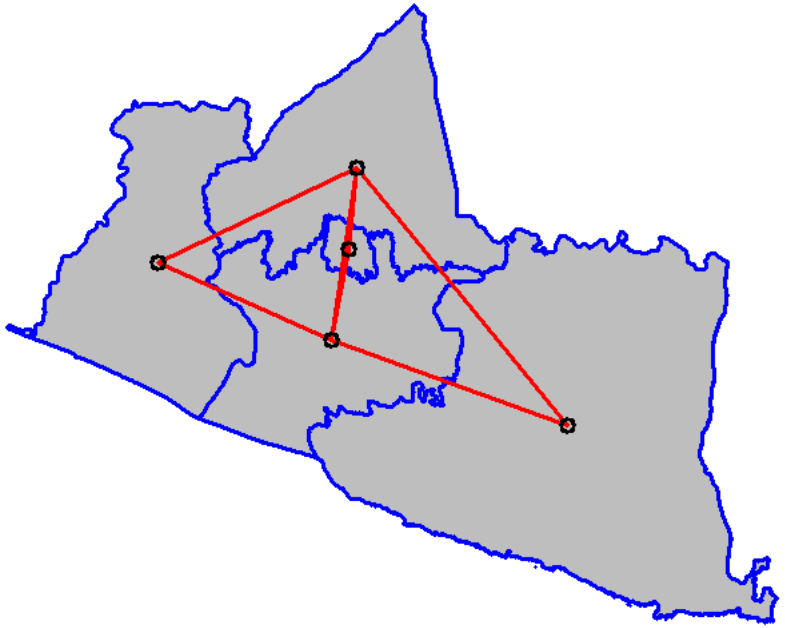
Aplikasi di R

Misalkan digunakan queen contiguity untuk pendefinisian tetangga

```
##tahap 2
yogya$id <- 0:4
library(raster)
plot(yogya, col=2:6)
text(yogya, 'id')

w <- poly2nb(yogya) #queen contiguity
plot(yogya, col='gray', border='blue', lwd=2)
plot(w, coords, col='red', lwd=2, add=TRUE)

###binary weights
w.bin <- nb2listw(w, style="B")
w.bin
w.bin.l<-nb2mat(w, style='B') #melihat matriks pembobot
w.bin.l
```

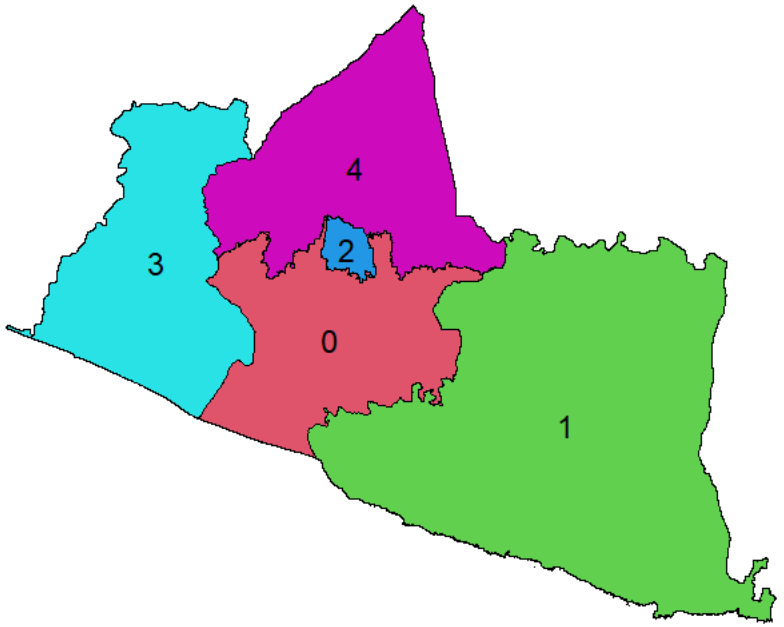
```
> w.bin.l
  [,1] [,2] [,3] [,4] [,5]
0    0    1    1    1    1
1    1    0    0    0    1
2    1    0    0    0    1
3    1    0    0    0    1
4    1    1    1    1    0
attr(,"call")
nb2mat(neighbours = w, style = "B")
```

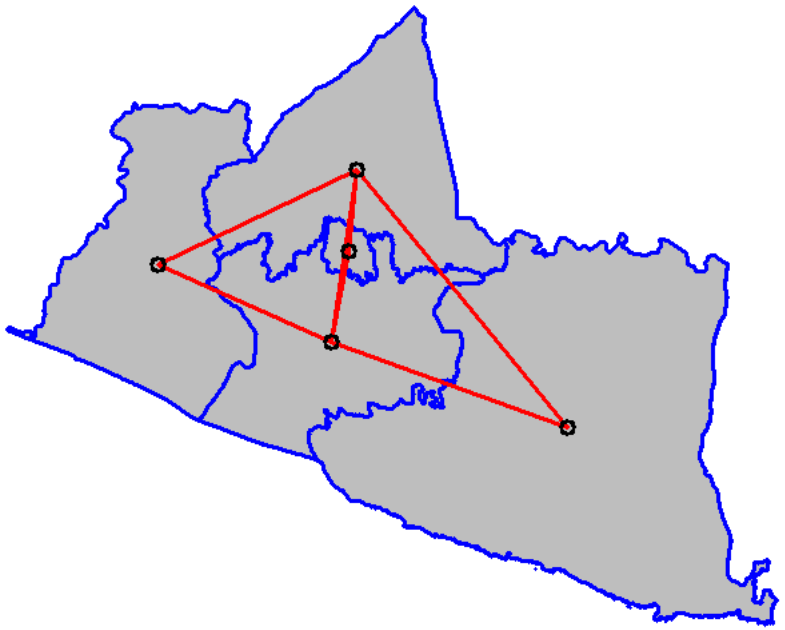




```
###row-standardized weights
w.rstd <- nb2listw(w)
w.rstd
w.rstd.l<-nb2mat(w) #melihat matriks pembobot
w.rstd.l
```

```
> w.rstd.l
  [,1] [,2] [,3] [,4] [,5]
0 0.00 0.25 0.25 0.25 0.25
1 0.50 0.00 0.00 0.00 0.50
2 0.50 0.00 0.00 0.00 0.50
3 0.50 0.00 0.00 0.00 0.50
4 0.25 0.25 0.25 0.25 0.00
attr(,"call")
nb2mat(neighbours = w)
```





Objek (area) tanpa tetangga

- Jika pernah mengelami error berikut

```
Error in nb2listw(filename): Empty neighbor sets found
```

- Berarti terdapat objek (area) yang tidak memiliki tetangga
- Maka perlu ditambahkan perintah ketika kasus ini terjadi

```
w.rstd <- nb2listw(w,zero.policy=T)
```

Tahapan 3

Spatial Autocorrelation

Uji autokorelasi spasial:

- Global
  - Moran's *I*
  - Geary's *C*
- Local
  - Local Moran's *I* (LISA – Local Indicators of Spatial Association)
  - Getis G_i^*

Moran's *I* is defined as

$$I = \frac{N}{W} \frac{\sum_i \sum_j w_{ij}(x_i - \bar{x})(x_j - \bar{x})}{\sum_i (x_i - \bar{x})^2}$$

where N is the number of spatial units indexed by i and j; x is the variable of interest; $\bar{x}$ is the mean of x; w_{ij} is a matrix of spatial weights with zeroes on the diagonal (i.e., $w_{ii} = 0$); and W is the sum of all w_{ij}.

Aplikasi di R

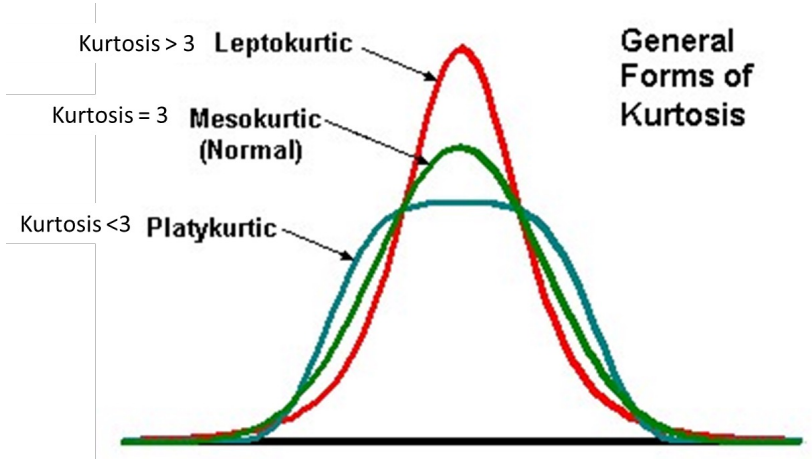
Misalkan digunakan queen contiguity untuk pendefinisian tetangga, dan digunakan binary weights matrix

```
###Moran Index
w <- poly2nb(yogya) #queen contiguity
w.bin <- nb2listw(w, style="B") #binary weights

moran(yogya$stunting,w.bin,n=length(w.bin$neighbours),S0=Szero(w.bin))
```

```
> moran(yogya$stunting,w.bin,n=length(w.bin$neighbours),S0=Szero(w.bin))
$I
[1] -0.1739506

$K
[1] 2.182911
```

Moran's I

sample kurtosis of x



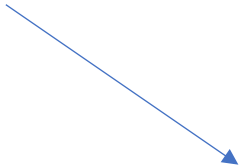
Moran Test

H_0: tidak terdapat autokoralasi spasial ($I = 0$)

H_1: terdapat autokoralasi spasial (two.sided → $I \neq 0$, greater → $I > 0$, less → $I < 0$)

```
###Moran Test
moran.test(yogya$stunting, w.bin, randomisation=F, alternative='two.sided')
```

```
> moran.test(yogya$stunting, w.bin, randomisation=F, alternative='two.sided')

        Moran I test under normality

data:  yogya$stunting
weights: w.bin

Moran I statistic standard deviate = 0.48851, p-value = 0.6252
alternative hypothesis: two.sided
sample estimates:
Moran I statistic       Expectation          Variance
      -0.17395062       -0.25000000        0.02423469
```



p-value > 0.05 → terima H_0

Local Moran's *I* (LISA – Local Indicators of Spatial Association)

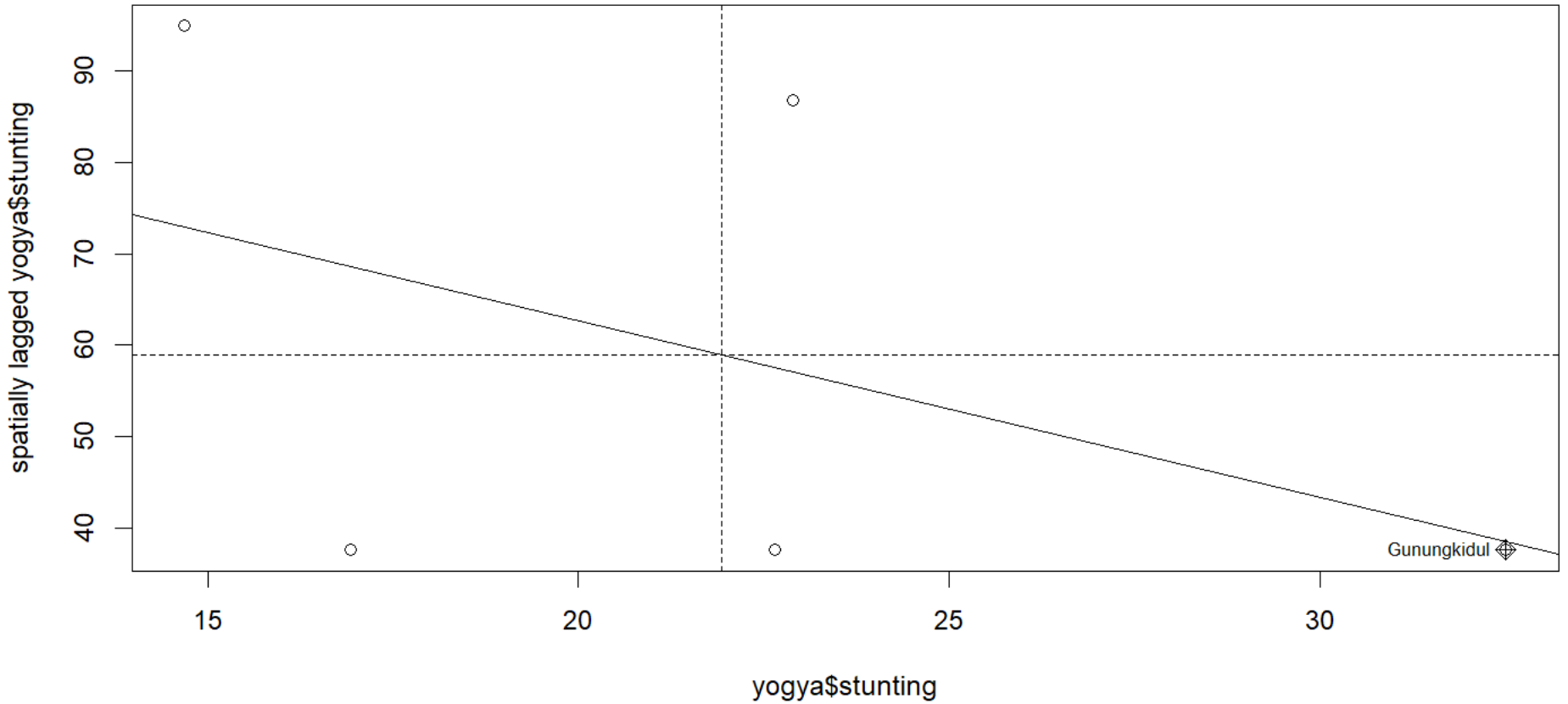
```
###Local Moran
localmoran(yogya$stunting, w.bin,alternative = 'greater')
```

```
> localmoran(yogya$stunting, w.bin,alternative = 'greater')
           Ii         E.Ii     Var.Ii       Z.Ii Pr(z > E(Ii))
0 -0.02395087 -0.023950868 0.00000000        Inf     0.0000000
1 -1.74276748 -1.465609806 1.30534585 -0.2425851     0.5958366
2  0.82458477 -0.328102908 0.91425716  1.2055284     0.1139997
3 -0.11798615 -0.006717405 0.02231614 -0.7448411     0.7718161
4 -1.37518889 -1.375188895 0.00000000       -Inf     1.0000000
attr(,"call")
localmoran(x = yogya$stunting, listw = w.bin, alternative = "greater")
attr(,"class")
[1] "localmoran" "matrix"     "array"
attr(,"quadr")
       mean    median    pysal
1 High-High High-High High-Low
2  High-Low  High-Low High-Low
3   Low-Low   Low-Low  Low-Low
4  High-Low   Low-Low High-Low
5  Low-High  Low-High Low-High
```

Diagram Pencar Moran

```
###Moran Scatter Plot
moran.plot(yogya$stunting, w.bin,labels=as.character(yogya$NAMOBJ))
```

labels are plotted for points with large influence



Terima kasih ☺