

# Boosting\_Contiguity\_Rook\_PDRB

Anandika Dian P.

3/29/2020

```
#Contiguity rook dengan ordo [1:3]  
#Matriks PDRB selisih terstandardisasi (nonsimetris)
```

```
start.time <- Sys.time()  
library(spdep)
```

```
## Loading required package: sp
```

```
## Loading required package: spData
```

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

```
## Loading required package: sf
```

```
## Linking to GEOS 3.6.1, GDAL 2.2.3, PROJ 4.9.3
```

```
dataIGI <- read.csv("D:/Skripsweet/data_jateng_w_latlong.csv")  
rownames(dataIGI) <- dataIGI$KABUPATEN  
attach(dataIGI)  
vy <- IGI  
vx <- dataIGI[,4:8]  
coords <- cbind(x_long,y_lat)  
detach(dataIGI)
```

```
PDRB <- read.csv("D:/Skripsweet/Kak Along's/matriks_PDRB_Berlaku_inverse.csv")  
w_PDRB <- as.matrix(PDRB)
```

```
library(rgdal)
```

```
## rgdal: version: 1.4-3, (SVN revision 828)  
## Geospatial Data Abstraction Library extensions to R successfully loaded  
## Loaded GDAL runtime: GDAL 2.2.3, released 2017/11/20  
## Path to GDAL shared files: C:/Program Files/R/R-3.6.2/library/rgdal/gdal  
## GDAL binary built with GEOS: TRUE  
## Loaded PROJ.4 runtime: Rel. 4.9.3, 15 August 2016, [PJ_VERSION: 493]  
## Path to PROJ.4 shared files: C:/Program Files/R/R-3.6.2/library/rgdal/proj  
## Linking to sp version: 1.3-1
```

```
jt_lp<-readOGR("D:/KULIAH STIS/Tingkat 3/Semester 6/SIG/UAS SIG","Jawa_Tengah")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "D:\KULIAH STIS\Tingkat 3\Semester 6\SIG\UAS SIG", layer: "Jawa_Tengah"
## with 35 features
## It has 6 fields
```

```
get.list=function(coords,k)
{
  #membuat ketetanggaan contuiguity rook
  nb<-poly2nb(jt_lp,queen = FALSE)
  nb_1<-nblag(nb,k)
  nb_cum<-nblag_cumul(nb_1)
  w_rook<-nb2listw(nb_cum,style = "B")
  #mengembalikan ke bentuk matriks
  w_rook <- listw2mat(w_rook)
  #mengalikan matriks knn dengan PDRB
  m_cust <- w_PDRB*w_rook
  m_cust_tot<-rowSums(m_cust,na.rm = TRUE)
  m_custom<-m_cust/m_cust_tot
  #mengubah ke bentuk listw
  w_custom <- mat2listw(m_custom)
}
```

```
create.instr=function(vy,vx,w_custom){
  ly <- lag.listw(w_custom,vy)
  res <- matrix(data = NA, nrow = nrow(vx), ncol = ncol(vx))
  for (i in 1:ncol(vx))
  {
    res[,i]= lag.listw(w_custom,vx[,i])
  }
  instr=lm(ly~ res)$fitted.values
  instr
}
```

```
#membuat ketetanggaan contuiguity rook
nb<-poly2nb(jt_lp,queen = FALSE)
w_q1<-nb2listw(nb,style = "B")
#mengembalikan ke bentuk matriks
w_q1 <- listw2mat(w_q1)
#mengalikan matriks knn dengan PDRB
m_cust_1 <- w_PDRB*w_q1
m_cust_1_tot<-rowSums(m_cust_1,na.rm = TRUE)
m_custom_1<-m_cust_1/m_cust_1_tot
#mengubah ke bentuk listw
w_custom_1 <- mat2listw(m_custom_1)
s12_o1<-create.instr(vy,vx,w_custom_1)
```

```
degs <- c(2,3)
res<-matrix(NA,length(vy),length(degs))
X <- vx
for(i in 1:length(degs))
{
  #a untuk membentuk matriks pembobot
  a <- get.list(coords,k=degs[[i]])
```

```

#menghitung fitted values model pada masing2 matriks pembobot
b <- create.instr(vy,vx,a)
names(b)=NULL
res[,i]=b
}
mn <- paste("s12_o",degs,sep="")
res <- data.frame(res)
names(res)<- mn
res<-cbind(s12_o1,res)
res<- as.matrix(res)
X <- cbind(X,res)
#X <- res

```

```

X1<-X[,-(1:5)]
n = nrow(X1)
#jumlah kombinasi matriks pembobot
#X berisi matriks dengan baris berupa wilayah dan kolom berupa macam2 matriks pembobot dengan elemen ma

save(X1,vy,n, file="d:/Skripsweet/Bimbingan 10/IGI_jateng.w12.Rdata")

load("d:/Skripsweet/Bimbingan 10/IGI_jateng.w12.Rdata")
jateng_w<-get(load("d:/Skripsweet/Bimbingan 10/IGI_jateng.w12.Rdata"))
jat_w_mat<-as.matrix(jateng_w)

```

```
library(mboost)
```

```
## Loading required package: parallel
```

```
## Loading required package: stabs
```

```
## This is mboost 2.9-1. See 'package?mboost' and 'news(package = "mboost")'
## for a complete list of changes.
```

```

#v 0.1
m1=glmboost(jat_w_mat,vy,control = boost_control(mstop = 1000,nu=0.1),center = FALSE)

```

```

aic1 <- AIC(m1,method="corrected")
aic1

```

```

## [1] -0.5637082
## Optimal number of boosting iterations: 1
## Degrees of freedom (for mstop = 1): 0.1

```

```

mbest1aic=m1[mstop(aic1)]
names(coef(mbest1aic)[abs(coef(mbest1aic)) > 0])

```

```
## [1] "s12_o2"
```

```
gMDL1 <- AIC(m1,method="gMDL")
gMDL1
```

```
## [1] -1.596576
## Optimal number of boosting iterations: 1
## Degrees of freedom (for mstop = 1): 0.1
```

```
mbest1gMDL=m1[mstop(gMDL1)]
names(coef(mbest1gMDL)[abs(coef(mbest1gMDL)) > 0])
```

```
## [1] "s12_o2"
```

```
end.time <- Sys.time()
time.taken <- end.time - start.time
time.taken
```

```
## Time difference of 9.132145 secs
```

```
nb<-poly2nb(jt_lp,queen = FALSE)
nb_1<-nblag(nb,2)
nb_cum<-nblag_cumul(nb_1)
w_rook<-nb2listw(nb_cum,style = "B")
#mengembalikan ke bentuk matriks
w_rook <- listw2mat(w_rook)
#mengalikan matriks knn dengan PDRB
m_cust <- w_PDRB*w_rook
m_cust_tot<-rowSums(m_cust,na.rm = TRUE)
m_custom<-m_cust/m_cust_tot
#mengubah ke bentuk listw
w_final <- mat2listw(m_custom,style = "W")
```

```
moran.test(dataIGI$IGI,w_final,randomisation = TRUE)
```

```
##
## Moran I test under randomisation
##
## data: dataIGI$IGI
## weights: w_final
##
## Moran I statistic standard deviate = 2.7973, p-value = 0.002577
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      0.27463248      -0.02941176      0.01181411
```

```
my<- dataIGI$IGI
mx<-cbind(dataIGI$INFLASI,dataIGI$PMTB,dataIGI$UMK,dataIGI$PP,dataIGI$PPS)
ols<-lm(my~mx)
moran.test(ols$residuals,w_final,randomisation = TRUE)
```

```
##
## Moran I test under randomisation
##
## data:  ols$residuals
## weights: w_final
##
## Moran I statistic standard deviate = 0.31645, p-value = 0.3758
## alternative hypothesis: greater
## sample estimates:
## Moran I statistic      Expectation      Variance
##      0.004573691      -0.029411765      0.011534112

r.ols <- lm(dataIGI$IGI ~ dataIGI$INFLASI+dataIGI$PMTB+dataIGI$UMK+dataIGI$PPS+dataIGI$PP)
r.lag <- lagsarlm(r.ols, dataIGI, w_final, zero.policy = TRUE)
```

```
## Warning: Function lagsarlm moved to the spatialreg package
```

```
## Registered S3 methods overwritten by 'spatialreg':
```

```
## method                from
## residuals.stsls        spdep
## deviance.stsls         spdep
## coef.stsls             spdep
## print.stsls            spdep
## summary.stsls          spdep
## print.summary.stsls    spdep
## residuals.gmsar        spdep
## deviance.gmsar         spdep
## coef.gmsar             spdep
## fitted.gmsar           spdep
## print.gmsar            spdep
## summary.gmsar          spdep
## print.summary.gmsar    spdep
## print.lagmess          spdep
## summary.lagmess        spdep
## print.summary.lagmess  spdep
## residuals.lagmess      spdep
## deviance.lagmess       spdep
## coef.lagmess           spdep
## fitted.lagmess         spdep
## logLik.lagmess         spdep
## fitted.SFResult        spdep
## print.SFResult         spdep
## fitted.ME_res          spdep
## print.ME_res           spdep
## print.lagImpact        spdep
## plot.lagImpact         spdep
## summary.lagImpact      spdep
## HPDinterval.lagImpact  spdep
## print.summary.lagImpact spdep
## print.sarlm            spdep
## summary.sarlm          spdep
## residuals.sarlm        spdep
## deviance.sarlm         spdep
```

```
## coef.sarlm          spdep
## vcov.sarlm          spdep
## fitted.sarlm        spdep
## logLik.sarlm        spdep
## anova.sarlm         spdep
## predict.sarlm       spdep
## print.summary.sarlm spdep
## print.sarlm.pred    spdep
## as.data.frame.sarlm.pred spdep
## residuals.spautolm  spdep
## deviance.spautolm   spdep
## coef.spautolm       spdep
## fitted.spautolm     spdep
## print.spautolm      spdep
## summary.spautolm    spdep
## logLik.spautolm     spdep
## print.summary.spautolm spdep
## print.WXImpact      spdep
## summary.WXImpact    spdep
## print.summary.WXImpact spdep
## predict.SLX         spdep
```

```
summary(r.lag, Nagelkerke = TRUE)
```

```
##
## Call:spatialreg::lagsarlm(formula = formula, data = data, listw = listw,
##   na.action = na.action, Durbin = Durbin, type = type, method = method,
##   quiet = quiet, zero.policy = zero.policy, interval = interval,
##   tol.solve = tol.solve, trs = trs, control = control)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.945608 -0.205644  0.026221  0.228951  0.518116
##
## Type: lag
## Coefficients: (asymptotic standard errors)
##              Estimate Std. Error z value Pr(>|z|)
## (Intercept)    5.3487367  1.2058107  4.4358 9.173e-06
## dataIGI$INFLASI -0.2001861  0.1189617 -1.6828 0.092418
## dataIGI$PMTB    0.0105209  0.0045979  2.2882 0.022127
## dataIGI$UMK     -0.0856851  0.0385487 -2.2228 0.026231
## dataIGI$PPS     0.3370628  0.1124002  2.9988 0.002711
## dataIGI$PP      -0.0042461  0.0062544 -0.6789 0.497203
##
## Rho: 0.29294, LR test value: 2.3414, p-value: 0.12597
## Asymptotic standard error: 0.1812
## z-value: 1.6167, p-value: 0.10595
## Wald statistic: 2.6136, p-value: 0.10595
##
## Log likelihood: -7.736404 for lag model
## ML residual variance (sigma squared): 0.089307, (sigma: 0.29884)
## Nagelkerke pseudo-R-squared: 0.53481
## Number of observations: 35
## Number of parameters estimated: 8
```

```
## AIC: 31.473, (AIC for lm: 31.814)
## LM test for residual autocorrelation
## test value: 7.5744, p-value: 0.0059204
```

```
W <- as(w_final, "CsparseMatrix")
trMatc <- trW(W, type = "mult")
```

```
## Warning: Function trW moved to the spatialreg package
```

```
set.seed(1)
summary(impacts(r.lag, tr=trMatc, R = 99), zstats = TRUE, short = TRUE)
```

```
## Warning: Method impacts.sarlm moved to the spatialreg package
```

```
## Impact measures (lag, trace):
##               Direct      Indirect      Total
## dataIGI$INFLASI -0.204355756 -0.078770499 -0.283126255
## dataIGI$PMTB    0.010740003  0.004139817  0.014879820
## dataIGI$UMK     -0.087469872 -0.033715935 -0.121185808
## dataIGI$PPS     0.344083450  0.132629614  0.476713064
## dataIGI$PP      -0.004334507 -0.001670769 -0.006005276
## =====
## Simulation results (asymptotic variance matrix):
## =====
## Simulated standard errors
##               Direct      Indirect      Total
## dataIGI$INFLASI 0.117683401 0.076336997 0.164835240
## dataIGI$PMTB    0.004754084 0.003982939 0.007405763
## dataIGI$UMK     0.039268048 0.039806418 0.069876912
## dataIGI$PPS     0.124631992 0.113668439 0.200204149
## dataIGI$PP      0.006375893 0.003583551 0.009277626
##
## Simulated z-values:
##               Direct      Indirect      Total
## dataIGI$INFLASI -1.8285605 -1.0425769 -1.7883215
## dataIGI$PMTB    2.1345597  0.9869931  1.9010883
## dataIGI$UMK     -2.2249290 -0.9263567 -1.7780345
## dataIGI$PPS     2.7343844  1.2129464  2.3908870
## dataIGI$PP      -0.6329746 -0.5606633 -0.6515615
##
## Simulated p-values:
##               Direct      Indirect      Total
## dataIGI$INFLASI 0.0674655 0.29714 0.073724
## dataIGI$PMTB    0.0327970 0.32365 0.057290
## dataIGI$UMK     0.0260860 0.35426 0.075398
## dataIGI$PPS     0.0062497 0.22515 0.016808
## dataIGI$PP      0.5267502 0.57503 0.514684
```