**##############################\**

# getting working directory & reading required packeges

getwd()

library(gtools)

**# defining the data set**

dane<-read.csv("example.csv", header=TRUE, dec=",", sep=";")

sum.by.regions<-dane[5, 2:8]

sum.by.sectors<-dane[1:4, 8]

sectors<-dane[1:4,1]

empl<-dane[1:4, 2:7]

m<-6 # numer of regions

n<-4 # numer of sectors

dane # displaying dataset - employment

# X region1 region2 region3 region4 region5 region6 Total

# Industry1 1 11 21 70 10 6 119

# Industry2 1 40 24 40 5 11 121

# Industry3 5 13 21 30 35 1 105

# Industry4 9 14 14 11 3 17 68

# Total 16 78 80 151 53 35 413

**##############################\**

**# defining the data on firms**

# distribution of firms by size

firmy<-read.csv("firms.csv", header=TRUE, dec=",", sep=";")

dim(firmy)

firmy # displaying dataset - firms

# sector size100 size50 size10 size1 Total

#Industry1 0 1 6 9 119

#Industry2 0 0 11 11 121

#Industry3 0 0 9 15 105

#Industry4 0 0 4 28 68

**# matrix of size of companies**

size<-matrix(0, nrow=dim(firmy)[1], ncol=dim(firmy)[2]-2)

size[,1]<-rep(100, times= dim(firmy)[1])

size[,2]<-rep(50, times= dim(firmy)[1])

size[,3]<-rep(10, times= dim(firmy)[1])

size[,4]<-rep(1, times= dim(firmy)[1])

size

# [,1] [,2] [,3] [,4]

#[1,] 100 50 10 1

#[2,] 100 50 10 1

#[3,] 100 50 10 1

#[4,] 100 50 10 1

**##############################\**

**# defining the distance between regions - przykładowe**

dist<-read.csv("dist.csv", header=TRUE, dec=",", sep=";")

dim(dist)

colnames(dist)<-paste("region",1:m)

rownames(dist)<-paste("region",1:m)

dist

# region 1 region 2 region 3 region 4 region 5 region 6

#region 1 0.000 0.781 0.403 0.430 0.540 0.549

#region 2 0.781 0.000 0.853 0.537 0.289 0.527

#region 3 0.403 0.853 0.000 0.318 0.727 0.858

#region 4 0.430 0.537 0.318 0.000 0.459 0.657

#region 5 0.540 0.289 0.727 0.459 0.000 0.266

#region 6 0.549 0.527 0.858 0.657 0.266 0.000

**# defining the map for spatial weights matrix**

library(spdep)

library(maptools)

library(sp)

library(rgdal)

map<-readOGR(".", "map6selected")

projekcja<-"+proj=longlat +datum=WGS84"

map<-spTransform(map, CRS(projekcja))

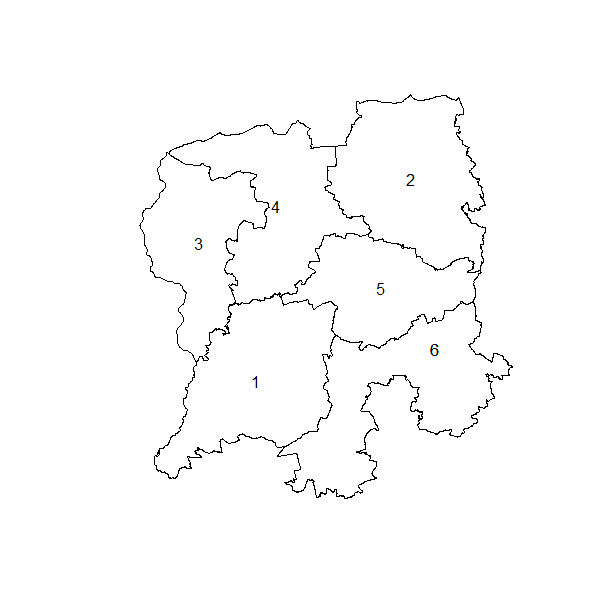
plot(map)

crdsb<-coordinates(map)

b<-1:6

text(crdsb[1:5,], labels=b[1:5])

text(15.7, 52.1, labels=b[6])



**# defining the spatial weights matrix**

cont.nb<-poly2nb(as(map, "SpatialPolygons"))

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

cont.mat<-nb2mat(cont.nb)

cont.mat

# [,1] [,2] [,3] [,4] [,5] [,6]

#0 0.00 0.00 0.25 0.25 0.25 0.25

#1 0.00 0.00 0.00 0.50 0.50 0.00

#2 0.50 0.00 0.00 0.50 0.00 0.00

#3 0.25 0.25 0.25 0.00 0.25 0.00

#4 0.25 0.25 0.00 0.25 0.00 0.25

#5 0.50 0.00 0.00 0.00 0.50 0.00

**##############################\**

**# sectoral concentration – measures by sectors for regions**

**##############################\**

**# object for all results together**

wyniki.all<-matrix(0, ncol=m, nrow=13)

rownames(wyniki.all)<-c("Ogive index", "Krugman dissimilarity index", "National Averages Index (NAI)", "Hallet index", "Relative Diversity Index (RDI)", "Shannon’s H", "Relative H", "Theil’s H", "KLD", "index of diversification", "Hachman", "RSI (max LQ)", "Gini")

colnames(wyniki.all)<-paste("region",1:m)

**################################\**

**# common elements of calculations**

**# matrix of industrial shares by region**s

shares.in.columns<-matrix(0, ncol=m, nrow=n)

for(i in 1:n){

temp<-empl[i,]/sum.by.regions[1:m]

shares.in.columns[i,]<-as.matrix(temp[1,])}

colSums(shares.in.columns) # check of sums in columns

**# vector of industrial shares for national economy**

share.in.columns.extra<-sum.by.sectors/sum(sum.by.sectors)

**################################\**

**# Ogive index**

wynik<-matrix(0, ncol=m, nrow=1)

colnames(wynik)<-paste("region",1:m)

rownames(wynik)<-c("Ogive index")

share\_star<-matrix(1/n, ncol=m, nrow=n)

diff.sq<-(shares.in.columns-share\_star)^2

diff.sq

for(j in 1:m){

wynik[1,j]<- sum(diff.sq[,j])/(1/n)}

wyniki.all[1,]<-wynik

################################\

**# Krugman dissimilarity index, NAI, Hallet, Relative Diversity Index**

wynik<-matrix(0, ncol=m, nrow=4)

colnames(wynik)<-paste("region",1:m)

rownames(wynik)<-c("Krugman dissimilarity index", "National Averages Index (NAI)", "Hallet index", "Relative Diversity Index (RDI)")

for(i in 1:m){

diff.abs<-abs(shares.in.columns[,i]- share.in.columns.extra)

wynik[1,i]<-sum(diff.abs)}

for(i in 1:m){

diff.abs<-(shares.in.columns[,i]- share.in.columns.extra)^2/share.in.columns.extra

wynik[2,i]<-sum(diff.abs)}

wynik[3,]<-wynik[1,]/2

wynik[4,]<-1/wynik[1,]

wyniki.all[2:5,]<-wynik

################################\

**# Entropy – Theil’s H, Shannon’s H, Relative H**

wynik<-matrix(0, ncol=m, nrow=4)

colnames(wynik)<-paste("region",1:m)

rownames(wynik)<-c("Shannon’s H", "Relative H", "Theil’s H", "KLD")

logs.matrix<-matrix(0, ncol=m, nrow=n)

logs.matrix<-shares.in.columns\*log(shares.in.columns)

logs.matrix

sum.columns<-colSums(logs.matrix)

sum.columns

logs.matrix.kld<-matrix(0, ncol=m, nrow=n)

logs.matrix.kld<-shares.in.columns\* log(shares.in.columns/ share.in.columns.extra)

sum.columns.kld<-colSums(logs.matrix.kld)

equal.dist.H<-abs((1/n)\*log(1/n)\*n)

wynik[1,]<-abs(sum.columns)

wynik[2,]<-abs(sum.columns)/rep(equal.dist.H, times=m)

wynik[3,]<-(-abs(sum.columns))+rep(equal.dist.H, times=m)

wynik[4,]<-abs(sum.columns.kld)

wyniki.all[6:9,]<-wynik

################################\

**# diversification index**

wynik<-matrix(0, ncol=m, nrow=1)

colnames(wynik)<-paste("region",1:m)

rownames(wynik)<-c("index of diversification")

hyp.con<-c(100, rep(0, times=n-1))

for(j in 1:m){

vec1<- shares.in.columns[,j]\*100

vec2<- share.in.columns.extra\*100

vec1.s<-sort(vec1, decreasing=TRUE)

vec2.s<-sort(vec2, decreasing=TRUE)

vec1.sc<-cumsum(vec1.s)

vec2.sc<-cumsum(vec2.s)

vec3.sc<-cumsum(hyp.con)

vec1.sum<-sum(vec1.sc)

vec2.sum<-sum(vec2.sc)

vec3.sum<-sum(vec3.sc)

index<-(vec1.sum- vec2.sum)/( vec3.sum- vec2.sum)

wynik[1,j]<-index}

wyniki.all[10,]<-wynik

**################################\**

**# LQ, Hachman, RSI**

wynik<-matrix(0, ncol=m, nrow=2)

colnames(wynik)<-paste("region",1:m)

rownames(wynik)<-c("Hachman", "RSI (max LQ)")

LQ<-matrix(0, ncol=m, nrow=n)

rownames(LQ)<-sectors

for(j in 1:m){

vec1<- shares.in.columns[,j]

vec2<- share.in.columns.extra

LQ[,j]<-vec1/vec2}

temp<-LQ\*shares.in.columns

temp1<-colSums(temp)

wynik[1,]<-1/temp1

wynik[2,]<-apply(LQ, 2, max) # max po kolumnach

wyniki.all[11:12,]<-wynik

**################################\**

**# Gini**

wynik<-matrix(0, ncol=m, nrow=1)

colnames(wynik)<-paste("region",1:m)

rownames(wynik)<-c("Gini")

for(j in 1:m){

vec1<- shares.in.columns[,j]

vec2<- share.in.columns.extra

vec3<-vec1/vec2

vec3.av<-mean(vec3)

vec3.sort<-sort(vec3, decreasing=TRUE)

vec4<-1:n

vec5<-abs(vec3.sort-vec3.av)\*vec4

tot.vec5<-sum(vec5)

wynik[1,j]<-(tot.vec5\*2)/(n\*n\* vec3.av)}

wyniki.all[13,]<-wynik

**################################\**

**wyniki.all**

**round(LQ,3)**

**################################\**

**# geographical concentration – measures by regions for sectors**

**##############################\**

**# object for all the results together**

wyniki.all<-matrix(0, ncol=n, nrow=14)

rownames(wyniki.all)<-c("Gini", "Krugman concentration index", "Bruelhart & Traeger", "locational Gini", "Agglomeration V", "Herfindahl", "Ellison-Glaeser", "Maurel-Sedillot", "Shannon’s H", "Relative H", "Theil’s H", "KLD", "Bergstrand clustering index","Moran’s I for LQ")

colnames(wyniki.all)<-paste("sector",1:n)

wyniki.all

shares.in.rows<-matrix(0, ncol=m, nrow=n)

for(i in 1:m){

temp<-empl[,i]/sum.by.sectors[1:n]

shares.in.rows[,i]<-as.matrix(temp)}

rowSums(shares.in.rows) # check of sums in columns

share.in.rows.extra<-sum.by.regions/sum(sum.by.regions[1:6])

shares.in.columns<-matrix(0, ncol=m, nrow=n)

for(i in 1:n){

temp<-empl[i,]/sum.by.regions[1:m]

shares.in.columns[i,]<-as.matrix(temp[1,])}

colSums(shares.in.columns) # check of sums in columns

share.in.columns.extra<-sum.by.sectors/sum(sum.by.sectors)

**################################\**

**# Gini**

wynik<-matrix(0, ncol=n, nrow=1)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Gini")

for(j in 1:n){

vec1<- shares.in.rows[j,]

vec2<- share.in.rows.extra[1:m]

vec3<-vec1/vec2

vec3.av<- rowMeans(vec3, na.rm=FALSE, dims=1)

vec3.sort<-sort(vec3, decreasing=TRUE)

vec3.sort

vec4<-1:m

vec5<-abs(vec3.sort-vec3.av)\*vec4

tot.vec5<-sum(vec5)

wynik[1,j]<-(tot.vec5\*2)/(m\*m\* vec3.av)}

wyniki.all[1,]<-wynik

**################################\**

**# Krugman concentration index**

wynik<-matrix(0, ncol=n, nrow=1)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Krugman concentration index")

for(i in 1:n){

diff.abs<-abs(shares.in.rows[i,]-share.in.rows.extra[,1:m])

wynik[1,i]<-sum(diff.abs)}

wyniki.all[2,]<-wynik

**################################\**

**# Bruelhart –Traeger index**

wynik<-matrix(0, ncol=n, nrow=1)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Bruelhart & Traeger index")

temp<-matrix(0, nrow=n, ncol=m)

for(i in 1:n){

temp[i,]<-as.matrix(shares.in.rows[i,]/share.in.rows.extra[,1:m])}

temp2<-matrix(0, nrow=n, ncol=m)

for(i in 1:n){

a<- share.in.rows.extra\*temp[i,]\*log(temp[i,], base=10)

temp2[i,]<-as.matrix(a[1,1:m])}

wynik<-rowSums(temp2)

wyniki.all[3,]<-wynik

**################################\**

**# locational Gini**

wynik<-matrix(0, ncol=n, nrow=1)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("locational Gini")

temp<-matrix(0, nrow=n, ncol=m)

for(i in 1:n){

temp[i,]<-as.matrix(shares.in.columns[i,]/share.in.rows.extra[,1:m])}

for(i in 1:n){

temp2<-matrix(0, nrow=m+1, ncol=m+1)

temp2[1,2:(m+1)]<-temp[i,]

temp2[2:(m+1),1]<-temp[i,]

a5<-mean(temp[i,])

for(g in 1:m){

for(h in 1:m){

temp2[g+1, h+1]<-abs(temp2[1,h+1]-temp2[g+1,1])}}

temp3<-temp2[2:(m+1), 2:(m+1)]

a3<-sum(temp3)

a4<-m\*(m-1)

G<-(a3/a4)/(n\*a5)

wynik[1,i]<-G}

wyniki.all[4,]<-wynik

**################################\**

**# Agglomeration V**

wynik<-matrix(0, ncol=n, nrow=1)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Agglomeration V")

temp1<-rowSums(shares.in.columns)/m

temp1a<-sum(share.in.rows.extra[1:m])/m

temp3<-matrix(0, ncol=m, nrow=n)

for(i in 1:m){

temp3[,i]<-(shares.in.columns[,i]-temp1)^2}

temp3<-as.data.frame(temp3)

temp3$total<-rowSums(temp3)

diff.share.row.sq<-(share.in.rows.extra[,1:m]-temp1a)^2

sum.diff.share.row.sq<-sum(diff.share.row.sq)

temp4<-(sum.diff.share.row.sq/m)^0.5

temp3$temp<-(temp3$total/m)^0.5

temp3$Vi<-(temp3$temp/temp1)/(temp4/temp1a)

wynik[1,]<-temp3$Vi

wyniki.all[5,]<-wynik[1,]

**################################\**

**# Herfindahl, Ellison-Glaeser**

wynik<-matrix(0, ncol=n, nrow=2)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Herfinahl", "Ellison-Glaeser")

temp<-matrix(0, nrow=dim(firmy)[1], ncol=dim(firmy)[2]-2)

for(i in 1: dim(firmy)[1]){

temp[,i]<-(size[,i]/firmy[,6])^2\*firmy[,i+1]}

temp<-as.data.frame(temp)

temp$total<-rowSums(temp)

wynik[1,]<- temp$total

temp<-matrix(0, nrow=n, ncol=m)

for(i in 1:n){

temp[i,]<-as.matrix((shares.in.rows[i,]-share.in.rows.extra[,1:m])^2)}

temp<-as.data.frame(temp)

temp$total<-rowSums(temp)

sum.share.row.sq<-sum(share.in.rows.extra[,1:m]^2)

eg<-matrix(0, nrow=n, ncol=3)

for(i in 1:n){

eg[,1]<-temp$total-(1-sum.share.row.sq)\*wynik[1,]

eg[,2]<-(1- sum.share.row.sq)\*(1-wynik[1,])

eg[,3]<-eg[,1]/eg[,2]}

wynik[2,]<-eg[,3]

wyniki.all[6:7,]<-wynik[1:2,]

**################################\**

**# Maurel-Sedillot index**

wynik<-matrix(0, ncol=n, nrow=2)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Maurel-Sedillot", "Herfindahl")

temp<-matrix(0, nrow=dim(firmy)[1], ncol=dim(firmy)[2]-2)

for(i in 1: dim(firmy)[1]){

temp[,i]<-(size[,i]/firmy[,6])^2\*firmy[,i+1]}

temp<-as.data.frame(temp)

temp$total<-rowSums(temp)

wynik[2,]<- temp$total

temp1<-shares.in.rows^2

temp1<-as.data.frame(temp1)

temp1$total<-rowSums(temp1)

temp1a<-share.in.rows.extra[1:m]^2

sum.temp1a<-sum(temp1a)

ms<-matrix(0, nrow=n, ncol=3)

for(i in 1:n){

ms[,1]<-(temp1$total-sum.temp1a)/(1-sum.temp1a)-wynik[2,]

ms[,2]<-(1-wynik[2,])

ms[,3]<-ms[,1]/ms[,2]}

wynik[1,]<-ms[,3]

wyniki.all[8,]<-wynik[1,]

**################################\**

**# Entropy – Theil’s H, Shannon’s H, Relative H, KLD**

wynik<-matrix(0, ncol=n, nrow=4)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Shannon’s H", "Relative H", "Theil’s H", "KLD")

logs.matrix<-matrix(0, ncol=m, nrow=n)

logs.matrix<-shares.in.rows\*log(shares.in.rows)

sum.rows<-rowSums(logs.matrix)

exp.share<-rep(1/m, times=m)

logs.matrix.kld<-matrix(0, ncol=m, nrow=n)

logs.matrix.kld<-shares.in.rows\*log(shares.in.rows/exp.share)

sum.rows.kld<-rowSums(logs.matrix.kld)

equal.dist.H<-abs((1/m)\*log(1/m)\*m)

wynik[1,]<-abs(sum.rows)

wynik[2,]<-abs(sum.rows)/rep(equal.dist.H, times=n)

wynik[3,]<-(-abs(sum.rows))+rep(equal.dist.H, times=n)

wynik[4,]<-abs(sum.rows.kld)

wyniki.all[9:12,]<-wynik

**################################\**

**# Clustering Index (Bergstrand)**

wynik<-matrix(0, ncol=n, nrow=1)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Bergstrand clustering index")

dist<-dist+0.001

for(i in 1:n){

temp2<-matrix(0, nrow=m+1, ncol=m+1) # counter

temp2[1,2:(m+1)]<-shares.in.columns[i,]

temp2[2:(m+1),1]<-shares.in.columns[i,]

for(g in 1:m){

for(h in 1:m){

temp2[g+1, h+1]<-(temp2[1,h+1]\*temp2[g+1,1]/dist[g,h])}}

temp3<-matrix(0, nrow=m+1, ncol=m+1) # nominator

temp3[1,2:(m+1)]<-as.matrix(share.in.rows.extra[1:m])

temp3[2:(m+1),1]<-as.matrix(share.in.rows.extra[1:m])

for(g in 1:m){

for(h in 1:m){

temp3[g+1, h+1]<-(temp3[1,h+1]\*temp3[g+1,1]/dist[g,h])}}

up<-sum(temp2[-1,-1])

down<-sum(temp3[-1,-1])

Bergs <-up/down

wynik[1,i]<-Bergs}

wyniki.all[13,]<-wynik

**################################\**

**# Moran for LQ**

wynik<-matrix(0, ncol=n, nrow=1)

colnames(wynik)<-paste("sector",1:n)

rownames(wynik)<-c("Moran’s I for LQ")

LQ<-matrix(0, ncol=m, nrow=n)

rownames(LQ)<-sectors

for(j in 1:m){

vec1<- shares.in.columns[,j]

vec2<- share.in.columns.extra

LQ[,j]<-vec1/vec2}

for(i in 1:n){

mm<-moran.test(LQ[i,], cont.listw)

wynik[1,i]<-mm$estimate[1]}

wyniki.all[14,]<-wynik

**################################\**

**# Arbia & Piras**

poss.rows<-20

wynik<-matrix(0, ncol=n, nrow=poss.rows)

colnames(wynik)<-paste("sector",1:n)

for(i in 1:n){

x<-as.matrix(dane[i,2:(m+1)])

a<-permutations(6, 6, v=x, set=FALSE, repeats.allowed=FALSE)

b<-matrix(0, nrow=dim(a)[1], ncol=2)

for(z in 1:dim(a)[1]){

mmm<-moran.test(a[z,], cont.listw)

b[z,1]<-as.numeric(mmm$estimate[1])

b[z,2]<-as.numeric(mmm$p.value)}

plot(density(b[,1]))

maks<-max(b[,1])

maks.list<-which(b[,1]==maks)

aver<-mean(a[1,])

down<-(dane[i,1:m+1]-rep(aver, times=m))^2

sum.down<-sum(down)

temp<-a[maks.list,]

temp2<-matrix(0, ncol=m, nrow=length(maks.list))

for(j in 1:length(maks.list)){

temp2[j,]<-as.matrix((temp[j,]-rep(aver, times=m))\*(dane[i,1:m+1]-rep(aver, times=m)))}

temp2<-as.data.frame(temp2)

temp2$sum<-rowSums(temp2)

temp2$lambda<-temp2$sum/sum.down

wynik[,i]<-c(temp2$lambda, rep(NA, times=poss.rows-dim(temp2)[1]))}

wynik

**################################\**

**wyniki.all**

**wynik**

**################################\**