Spatial Economics – Assignment 1

Gustav Pirich (h11742049@s.wu.ac.at) Gabriel Konecny (h11775903@s.wu.ac.at)

2024-03-18

Exercise A

For our independent variables, we use per capita crime rate by town, average number of rooms per dwelling, Charles River dummy variable (= 1 if tract bounds river; 0 otherwise), nitrogen oxides concentration (parts per 10 million) and a constant.

Below we wrote a function which computes: OLS point estimates for the intercept, slope parameters, and the error variance. Suitable test statistics with corresponding p-values for the relevant coefficients. Intervals of the coefficients for a confidence level of 95%.

```
OLS <- function(X,Y){
  beta_hat <- solve(t(X) %*% X) %*% t(X) %*% Y  # OLS estimates for coefficients
Y_hat <- X %*% beta_hat  # Fitted values
e <- Y - Y_hat  #residuals
n <- nrow(X)  # Number of observations
k <- ncol(X) - 1  # Number of covariates excluding intercept
s <- as.numeric(t(e)%*%e / (n-k)) #SSR adjusted for degrees of freedom
sigma <- s*solve(t(X) %*% X) #VCV of Beta hat
se <- sqrt(diag(sigma))
t_stat <- (beta_hat-0) / se
p <- pt(abs(t_stat), n-k, lower.tail=FALSE)
th <- qt(0.975, n-k)
conf <- cbind(beta_hat-th*se,beta_hat+th*se)
colnames(beta_hat) <- "estimate"
colnames(conf) <- c("2.5%","97.5%")
colnames(t_stat) <- "t-statistic"
colnames(p) <- "p-value"
error_variance <- s
list(rbind(beta_hat,error_variance), cbind(t_stat, p), conf)
}
```

The output of the function is presented below:

```
OLS(X,Y)
```

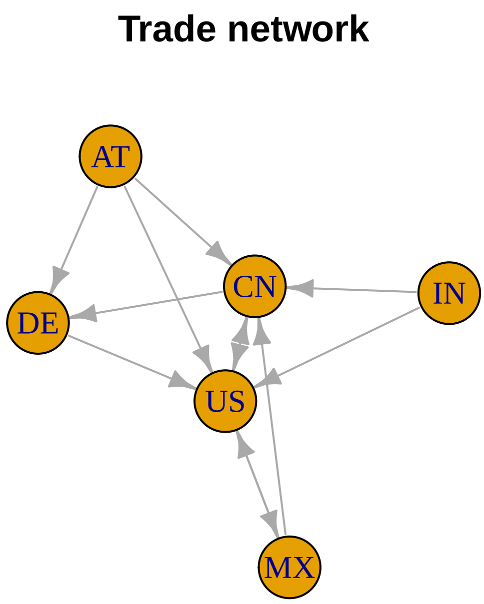
```
## [[1]]
##                  estimate
## Constant       -17.259625
## Crime           -0.184610
## Rooms            7.706840
## Charles_River    4.673807
## NO_pp10m       -14.960358
## error_variance  35.771378
##
## [[2]]
##               t-statistic      p-value
## Constant        -5.373025 5.935625e-08
## Crime           -5.358215 6.414580e-08
## Rooms           19.155659 4.233091e-62
## Charles_River    4.388052 6.974274e-06
## NO_pp10m        -5.674181 1.178869e-08
##
```

```
## [[3]]
##                      2.5%       97.5%
## Constant      -23.5707812 -10.9484684
## Crime          -0.2523012  -0.1169189
## Rooms           6.9163879   8.4972927
## Charles_River   2.5811627   6.7664511
## NO_pp10m      -20.1404236  -9.7802928
```

Exercise B

1. Draw a graph of the network; create the adjacency matrix in R.

We could consider 6 different countries and an indicator of a high trade between them. For illustration, we could consider a rule where there is directed edge from A to B, if B is one of the top 5 export destinations of A. We dont actually check for biggest trading partners empirically, but lets assume that such procedure gives rise to following network (arrows are mostly made up to achieve 10 edges and nice interpretations):



The corresponding adjacency matrix is:

```
##    US MX DE AT CN IN
## US  0  1  0  0  1  0
## MX  1  0  0  0  1  0
## DE  1  0  0  0  0  0
## AT  1  0  1  0  1  0
## CN  1  0  1  0  0  0
## IN  1  0  0  0  1  0
```

2. Who are the most and least central agents in the network? Name, explain, and try to quantify different notions of centrality.

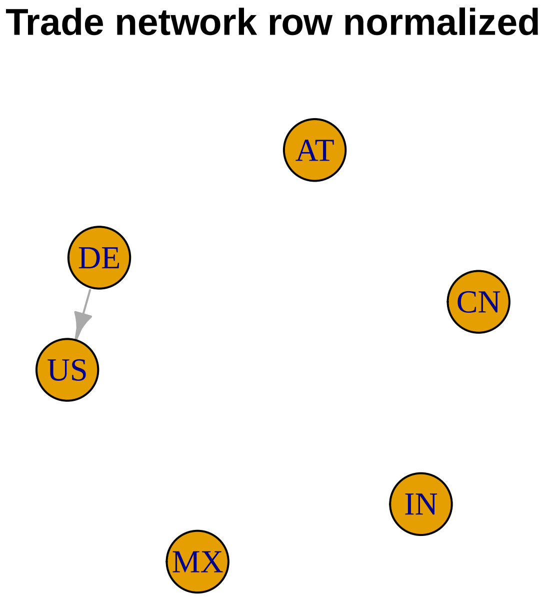
A very basic concept of centrality could define an agent as most central, if it has the highest number of directed edges pointing towards itself (i.e. if it is important export country for most other countries). Using this criterion we can see from the graph that or from columns of adjacency matrix that US the most central agents in this network, because it is top 5 trading partner for all 5 other countries. The least central agents would be Austria and India, since they are not a top 5 export country for any country from this network.

Another basic criterion of centrality could be the amount of outwards pointing arrows of an agent. Thus a country would be considered central if it exports to highest number of countries from this network. In this sense, Germany is least central with only 1 outward arrow, while Austria is most central with 3 outward arrows.

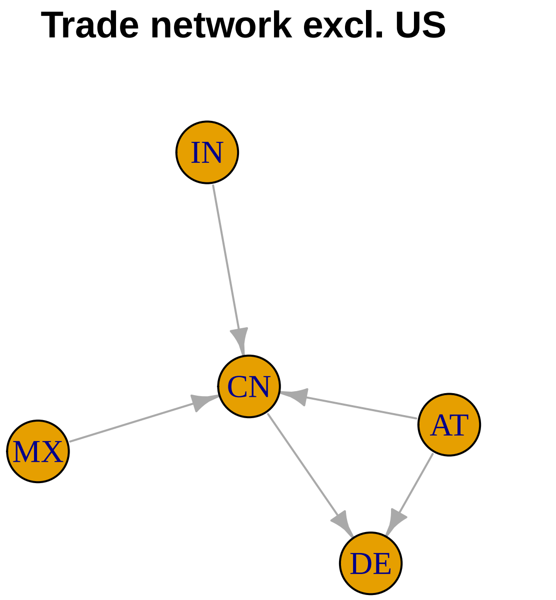
Eigenvector centrality: There is no sink, but not possible to get to Austria. ? Page Rank: weights (probs) need to be defined?

- How would centralities change if you considered a row-normalized network instead?

Germany which had only 1 connection, is the only row with sum of 1. Thus its the only edge which survives row normalization. Thus here depending on criterion US or Germany would be most central agent. Other countries would be all least central.



- How would the network change if you removed or added a specific agent? Lets consider the case of removing US:



Based on Inflowing arrows, China becomes the most central agent and India, Austria and Mexico the least central agents. Based on Outflowing arrows, Austria is still the most central agent, while Germany is the least central agent.

Exercise C

Download a suitable shapefile for NUTS2 regions (from here or using R directly) and some dataset of interest at the same level of aggregation (e.g. from here). Install and load the sf and ggplot2 packages together with their dependencies.

```
library(sf)
```

```
## Warning: package 'sf' was built under R version 4.3.1
```

```
## Linking to GEOS 3.11.0, GDAL 3.5.3, PROJ 9.1.0; sf_use_s2() is TRUE
```

```
library(ggplot2)
```

```r
# create a new empty object called 'temp' in which to store a zip file
# containing boundary data
temp <- tempfile(fileext = ".zip")
download.file("http://ec.europa.eu/eurostat/cache/GISCO/distribution/v2/nuts/download/ref-nuts-2021-03m

outDir <- "./data"
unzip(temp, exdir=outDir)

#list.files(path="./data", pattern = '*.shp')

# let us choose projection WGS 84 (EPSG 4326) which is visible the file name
# between the year (2021) and the level of the data (NUTS 2):
unzip("./data/NUTS_RG_03M_2021_4326_LEVL_2.shp.zip", exdir=outDir)
```

- Read in the shapefile, and find out what projection and CRS the file uses.

```r
shp <- st_read(dsn = "./data", layer ="NUTS_RG_03M_2021_4326_LEVL_2") #reads in the shapefile
```

```
## Reading layer `NUTS_RG_03M_2021_4326_LEVL_2' from data source
##   `/Users/gustavpirich/Library/Mobile Documents/com~apple~CloudDocs/Wirtschaftsuniversitaet/TUTOR/ad
##   using driver `ESRI Shapefile'
## Simple feature collection with 334 features and 9 fields
## Geometry type: MULTIPOLYGON
## Dimension:     XY
## Bounding box:  xmin: -63.15176 ymin: -21.38696 xmax: 55.83509 ymax: 80.83426
## Geodetic CRS:  WGS 84
```

```r
# The shapefile looks just like a normal dataframe with an additional "geometry" column:
#head(shp)
# Save crs for later
old_crs <- st_crs(shp)
#st_geometry(shp)
```

The projection used is WGS 84 (EPSG 4326).

Map the data to use another projection and/or CRS of your choosing: Since exclude everything except Austria, we decided for MGI Lambert Austria (equidistant equal-area specifically for AT).

```r
# we exclude everything except Austria
shp <- shp[shp$CNTR_CODE %in% "AT", ]
#Therefore we decide for MGI Lambert Austria (equidistant equal-area specifically for AT)
#plot(st_geometry(shp))
st_geometry(shp) <- st_geometry(st_transform(shp,
                  st_crs("epsg:31287")))
#plot(st_geometry(shp))
```