Exploring Socio-Economic and Health Deprivation in Edinburgh Data Zones, Using the Self-Organizing Maps

1. Introduction

This essay investigates the prevalence of socio-economic and health deprivation across 597 data zones within Edinburgh. The deprivation concept is used extensively not only in the analysis of social conditions but also, in an applied form, as an instrument for policy making in allocating resources to particular areas and services [1]. The association between socio-economic and health deprivation is a well-known phenomenon [2]. A data zone can be said to be in deprivation when its score in some of the socio-economic and health factors are seriously below the average, in a way that people living within that data zone are excluded from ordinary living patterns and activities [3]. In this essay, I used Self-Organizing Map (SOM) visualization technique to find out different levels of deprivation patterns and used clustering on SOMs to group similar nodes into one cluster. I trained my SOM model with two different datasets. First, I used all related variables (28 variables[1]) from the Scottish Index of Multiple Deprivation (SIMD 2016) excel file. Second, I trained the model with 10 variables with the highest variance. In both cases, I clustered the nodes using k-means clustering, and visualized the results by color coding a choropleth map. Also, I completed the manual identification of most-deprived clusters by exploring different types of SOMs and drawing up a "story" about different reasons of deprivation.

2. Data Procurement

I selected the data aligned with the Edinburgh boundary shapefile from the SIMD 2016 excel file (row number 1913 to 2509 - 597 columns in total) and stored it in a new csv file. The crime rate was negligible in some of the data zones and was indicated by * character in the source file, which I replaced with 0. Also, for some of the variables both count and rate were shown. In such cases, I chose the rate column. Table in the appendix section shows all the variables I saved in my newly made excel file.

3. Methodology

Understanding relationships in high-dimensional datasets could be difficult and requires proper data visualization techniques. Self-Organizing Map (SOM) is an unsupervised artificial neural network data visualization technique [4]. It is used to visualize high-dimensional data sets in a colorful 2D diagram of ordered nodes, containing similar samples [5]. The process of making a

[1]These variables will be mapped for the Data_Zone column. If a feature has both count and rate columns, the count column is excluded. Also, the Intermediate_Zone column and the Council_area column are excluded from these 28 columns of variables.

SOM involves the following steps: 1) choosing a representative subset of variables, standardizing[2] their values and making a matrix from these standardized values, 2) adjusting the size of the grid, 3) training the model, and 4) plotting the self-organizing map in one form of available plot types [6]. Note that each node vector has a fixed position on the SOM grid. To get the same results for different executions of the R code, I set a fixed random seed number (10), so that the first randomly allocated weights to grids remains the same.

3.1. Representative Variable Selection

As mentioned in the introduction section, I worked with two subsets of variables, hence two datasets. The first one contained every variable in table 1. The results from this dataset makes the comparison between SOM technique and SIMD map possible. The second dataset contains the 10 variables with highest variance (Crime_rate, Total_population,Working_age_population, Drug, Alcohol, CIF, Noquals, SMR, EMERG, PT_retail). Variables with highest variance are shown to be representative by statisticians [7]. In the following, I will only show the first dataset's plots to illustrate required adjustments for making SOMs (adjustments in the number of iterations (Section 3.2) and grid size (Section 3.3)); however, the same processes have been done for the second dataset. Then I will show various SOM types and their clusters for the both datasets in section 3.4.

3.2 Training the model

As the SOM training iterations progress, the distance from each node's weights to the samples represented by that node is reduced. Ideally, this distance should reach a minimum plateau. I tried different numbers and finally chose 1000 iterations, as the curve was not continually decreasing anymore (figure 1 below).

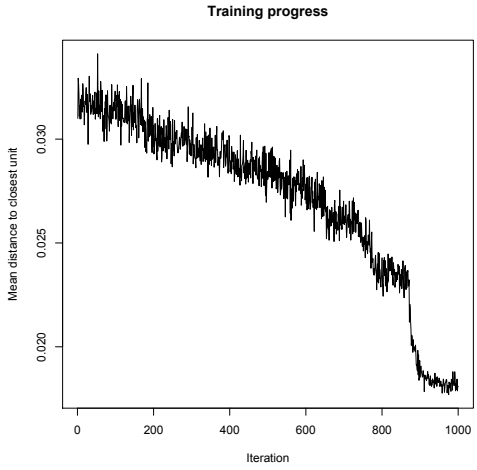


Figure 1. The training process for SOM model with the 1st dataset

[2] By calculating the z score

3.3 Choosing the grid size

For choosing the grid size, I considered the total number of records (597 records) and divided the number by 10 (ideally if sample distribution was relatively uniform amongst all nodes, I wanted to have 10 samples in each node), the quotient value gives a rough estimation of the optimal grid size. For a more accurate result I customized the grid size by applying different grid sizes and watching for positive or negative effects on node count and node quality SOMs. Figure 2 compares the SOM node count and quality for 8*8 and 9*9 grids. I selected 9*9 map size, since the quantity of samples in each node is closer to the recommended values (at least 5 to 10 samples per node [4]) and none of the nodes are over-populated or empty. Also, the node quality map in 9*9 grid size is higher. As in most of the nodes the average distance between the objects mapped to a node and the codebook vector of that node is small.

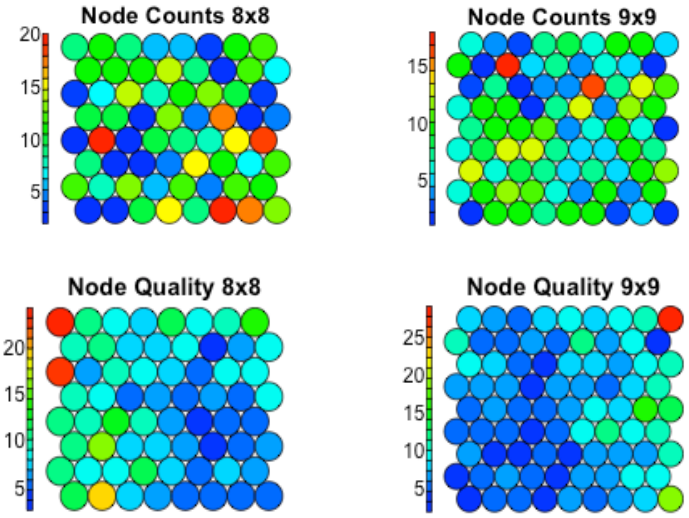


*Figure 2. Compression of node quality and node count in the 1st dataset for gird size 8*8 and 9*9*

3.4 Choosing the 'type' of plots for SOM model

The kohonen package implements several different types of plots for SOM modeling: Changes, Node Counts, Node Quality, Codes, Neighbor Distance, Weight Vectors and Heatmaps. Each one of these plots provide useful insights for a specific purpose [8]. In Figure 3 you can see the SOM codes and the 10 clusters[3] on it for the second dataset. I didn't include the SOM codes for the

[3] WCSS plots in appendix B explain why I chose 10 clusters for both dataset. 7 clusters also showed similar results.

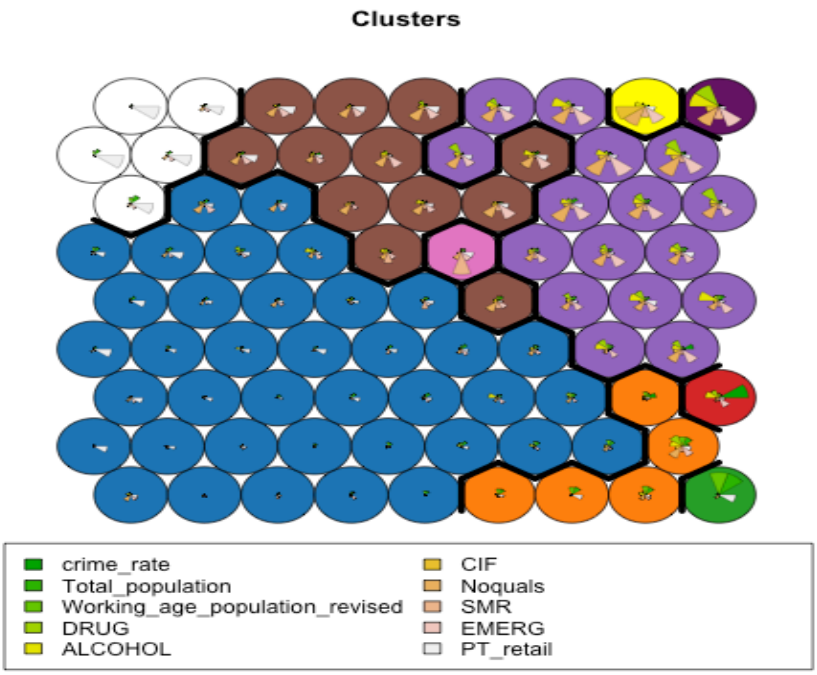
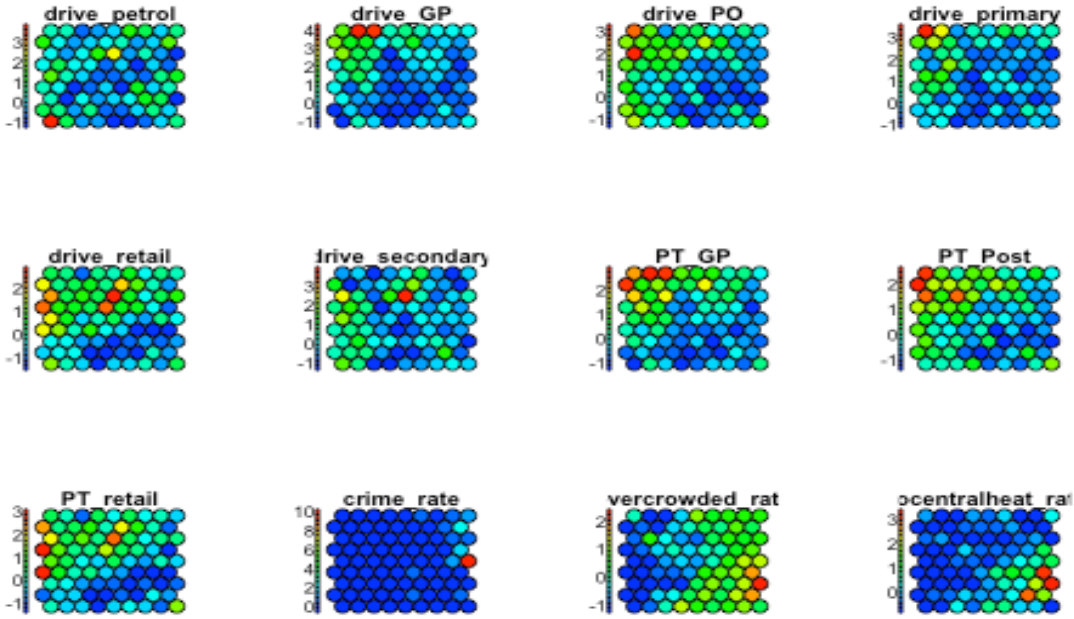


Figure 3. SOM codes for the 2nd dataset

first dataset in the report, as it was not helping with visualization due to its many variables. Codes visualizes the weight vectors on a "fan diagram" across the map, where we can see patterns in the distribution of samples and variables. It is notable that these clusters will retain their colors when transferring on the geographical Edinburgh shapefile in section 4.



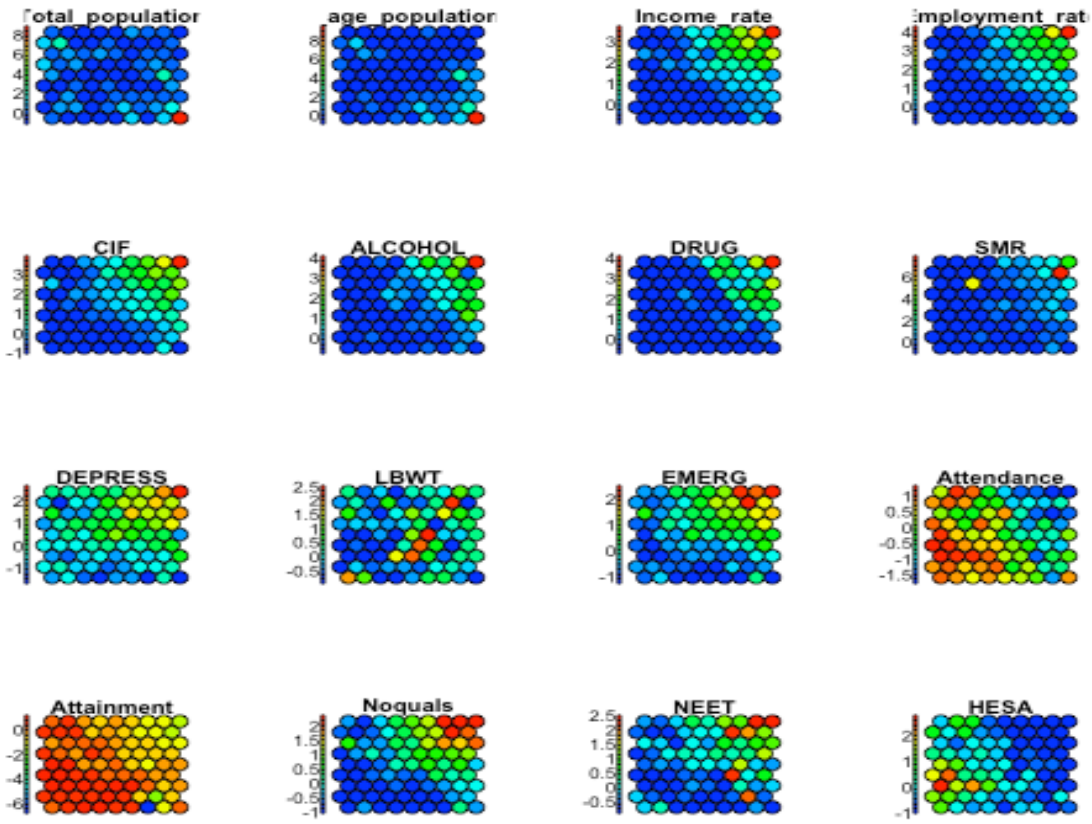


Figure 4. SOM heatmaps for the 28 variables in the first dataset

Heatmaps are perhaps the most important visualization possible for SOMs. The use of codes that tries to view all dimensions on the one diagram is unsuitable for high-dimensions (>7 variable). A SOM heatmap allows the visualizations of the distribution of a single variable across the map. It is important to remember that the individual sample positions do not move from one visualizations to another. Heatmaps for the 28 variables are shown in figure 4. As you can see there are relationships between some of these variables (please look at heatmaps of Drugs, Alcohol, Income rate, CIF, EMERG, NEET, etc.), as it is discussed in the existing literature. For example, there are relationships between alcohol misuse and major depression [9], effect of unemployment on crime [10] and depravation [11].

Figure 5 visualizes the distance between each node and its neighbors respectively for dataset 1 and 2 (left and right). In these neighbor distances maps, whenever a lighter node is adjacent to a darker one, a clustering has happened.

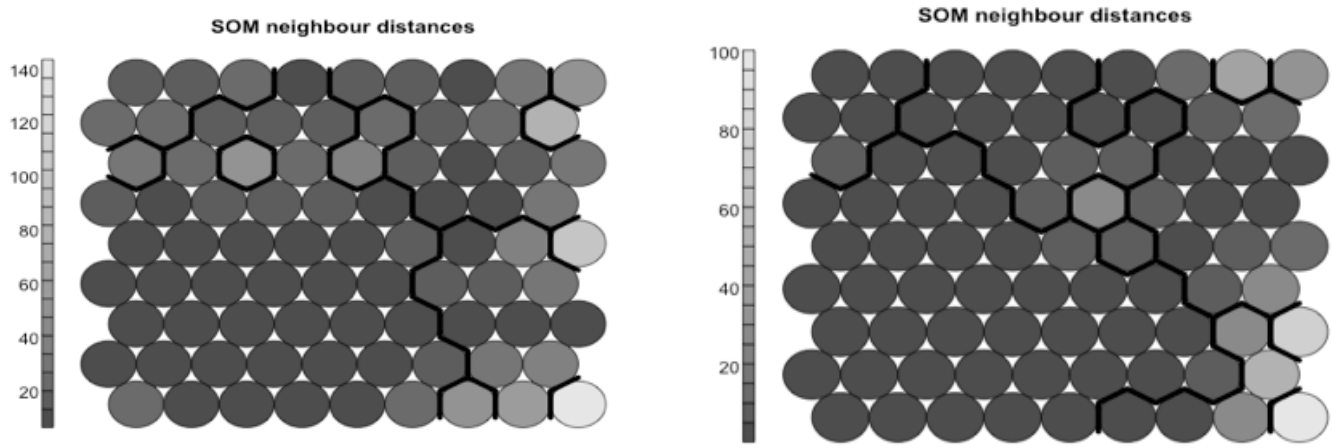


Figure 5. neighbor distance map for dataset 1 and 2 (left to right)

Figure 6 shows the different clusters respectively for dataset 1 and 2. I randomly assigned some place names to some nodes, so that anyone who knows one place might be able to get a better understanding for the whole cluster. In section 5, I related these map to the geographical area. Note that colors on bellow map and geographical maps in section 5 represent the same clusters.

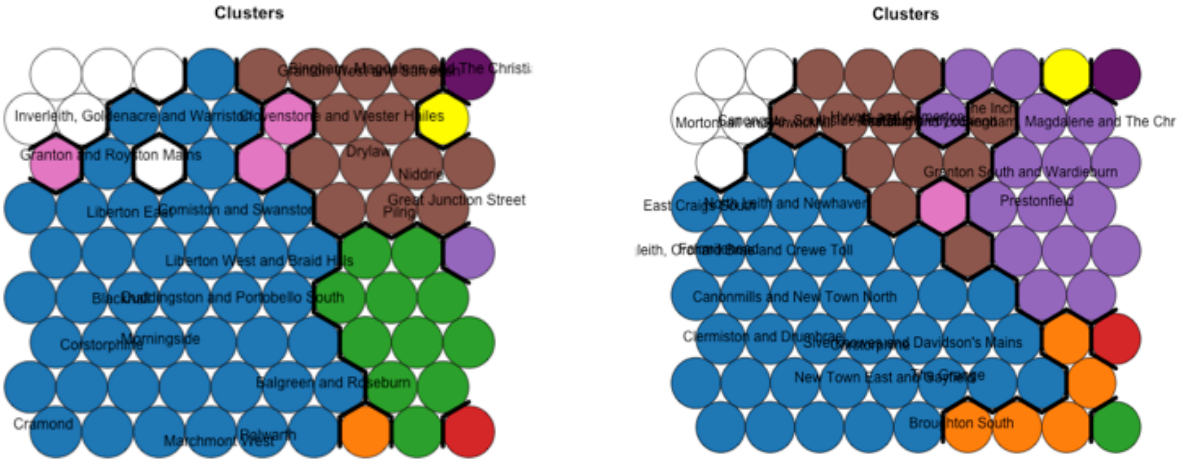


Figure 6. Labeled SOM grids with place name for dataset 1 and 2 (left to right)

4. Results and discussion

I transferred the clusters from SOMs to geographic maps (Figure 7 and 9). These figures respectively show the results for the first and second datasets. In the first look it can be seen that the two maps represent relatively similar clusters. This is good news, as it can be said that clustering has been done correctly. However, these clusters have no meaning in terms of whether an area on the map is least or most deprived or what is the living situation of this data zone in comparison to a different data zone in another cluster. The interpretation of these maps should be done considering other SOM-based plots and the following geographically-distributed single variable map (figure 8). The main SOM plots I used for interpretation of the first dataset are heatmaps (figure 4, section 3.4) and for the second dataset is the SOM codes plot (figure 3, section 3.4). I compared both geographically clustered maps to get a better sense of clusters.

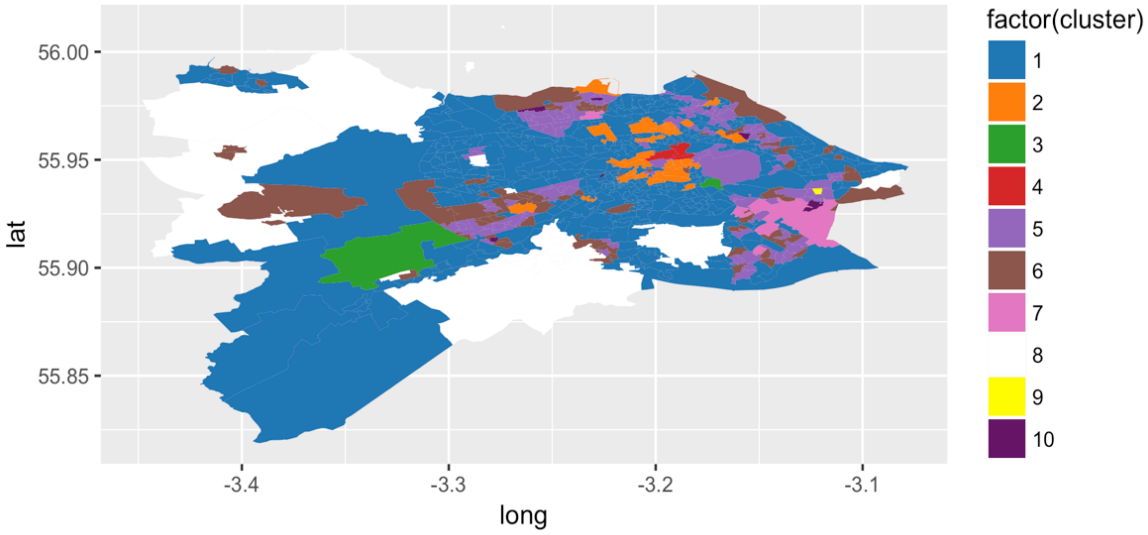
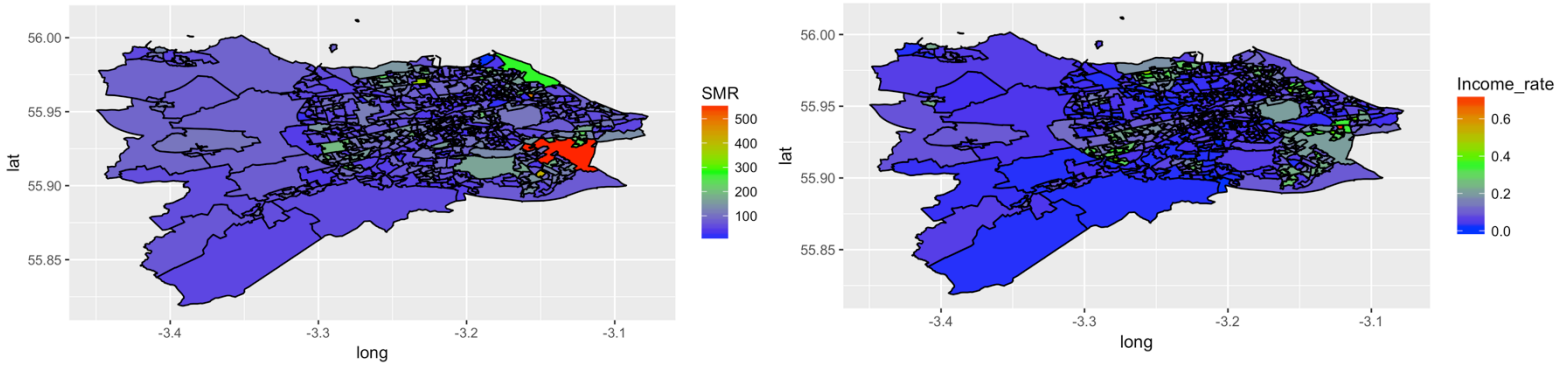


Figure 7. Edinburgh geographical map, containing 10 clusters of deprivation, using 28 variables



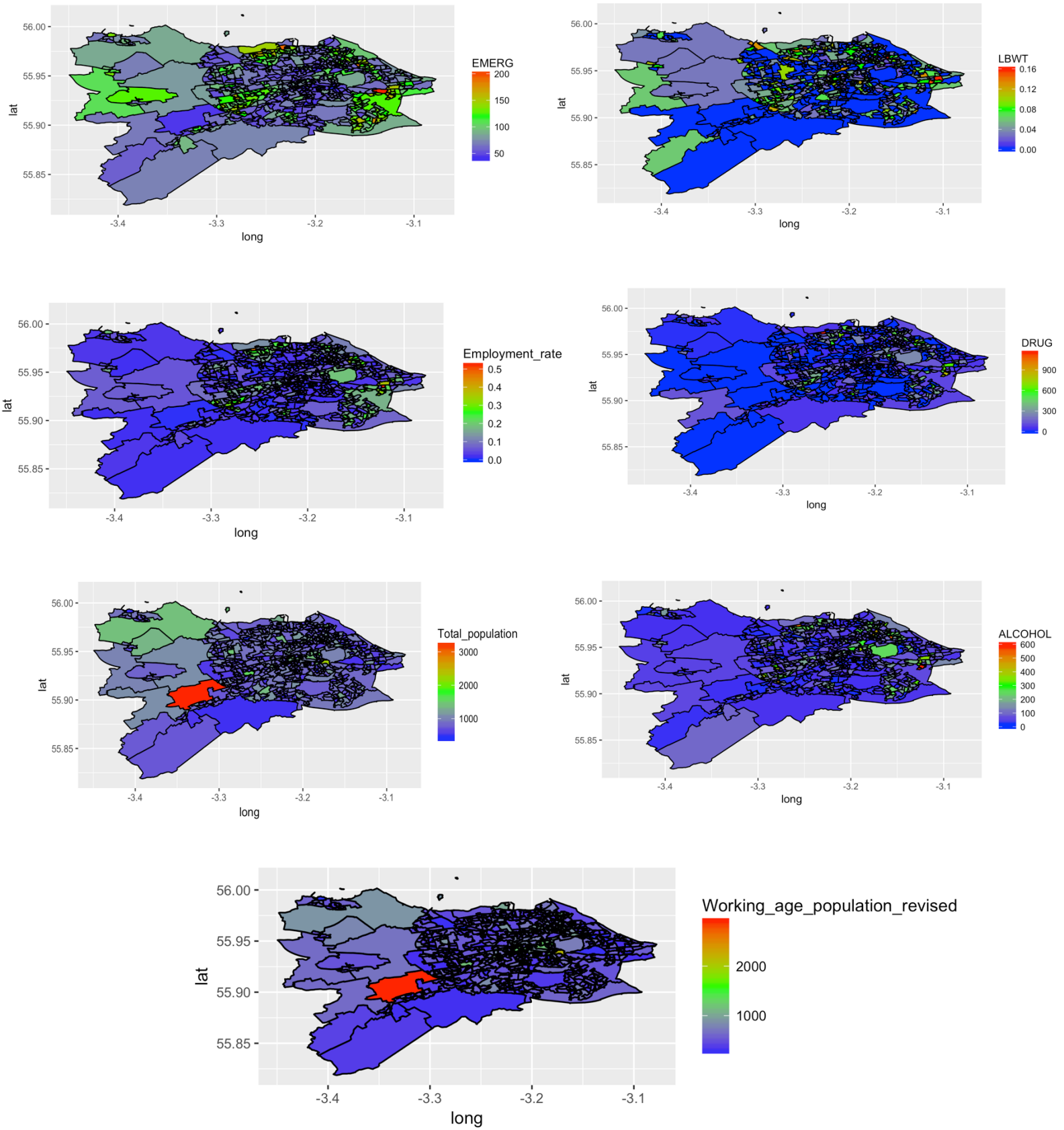


Figure 8. 28 single variables plotted on separate geographical maps

For the first dataset (figure 7), cluster 1 contains areas with least socio-economic depravation variables. On the other hand, clusters 5, 6 and 7 contain areas with multiple depravation variables; however, each one of these clusters have some patterns of depravation, which is different from the others. In particular, all three clusters have high rate of emergency stay in hospital. Additionally, cluster 7 is more deprived in terms of mortality, income rate and employment rate. Cluster 6 generally has high mortality and cluster 5 has low employment rate.

Cluster 3 probably has been created mainly because the data zones within it have very high total population and working age population. I did not exclude the total population and working age population from my datasets as more people in an area means less resources for each, also more working age people means less job opportunity for people within that area.

For the second dataset, its SOM code plot (figure 3, section 3.4) shows clearly how each cluster differs from others in terms of deprivation. The most deprived data zone is 5, then 6 and 9, which span areas relatively similar to the first dataset.

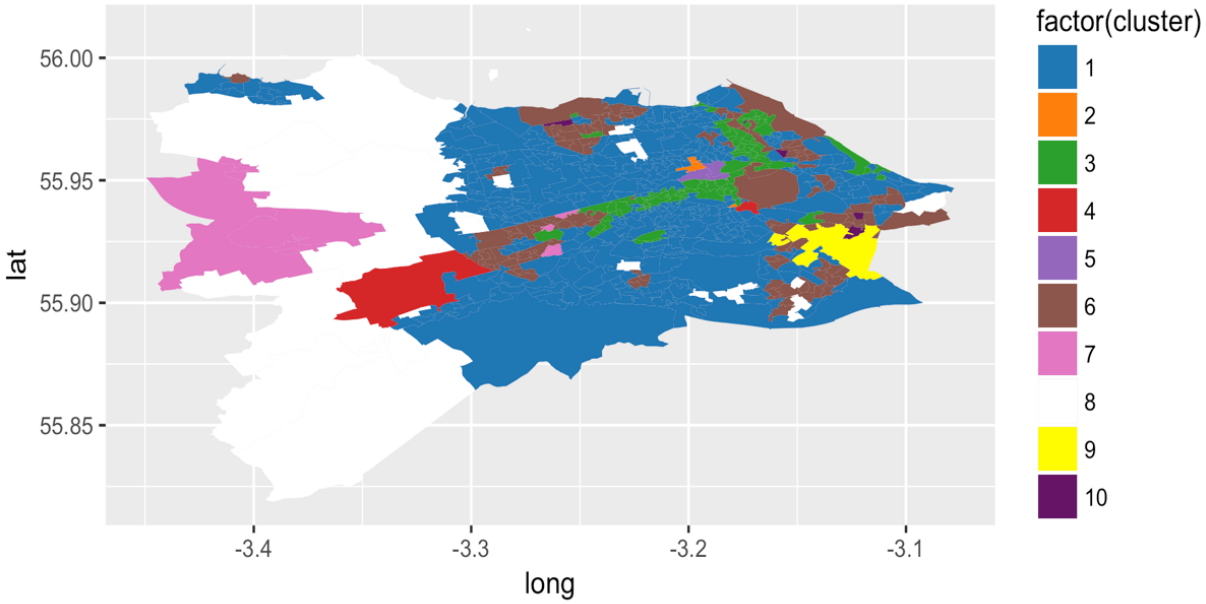


Figure 9. Edinburgh geographical map, containing 10 clusters of deprivation, using 10 variables

5. Suggestions and future works

Using choropleth maps might be misleading, as one can apply any kind of boundaries based on population or something else. In our case, one might note a big data zone area which is color coded in red but overlooks a small with the same color. However, the smaller area might be a populated zone with people who are living in very poor conditions. For future work, one might be interested in making SOMs just with variables related to a specific background. For example, it is interesting to investigate health amongst the data zones and only uses the health-related variables, instead of having a mixed combination of variables with different backgrounds.

6. Pros and Cons of using SOMs

FSOM is a powerful tool in data visualization. With its various types of plots, it is easy to explain results even to nonprofessionals. Also, it has relatively simple algorithm for clustering. However, it has its disadvantages. Its training requires clean, numeric data which can be hard to get. It can be difficult to represent many variables in two-dimensional plane. Its clusters have no order and they need manual identification for interpretation. With large number of clusters, it is even hard to identify what variables has caused generation of new clusters. Moreover, it is likely to give clusters with few members.

Appendix A: Description of the data file extracted from SIMD 2016.

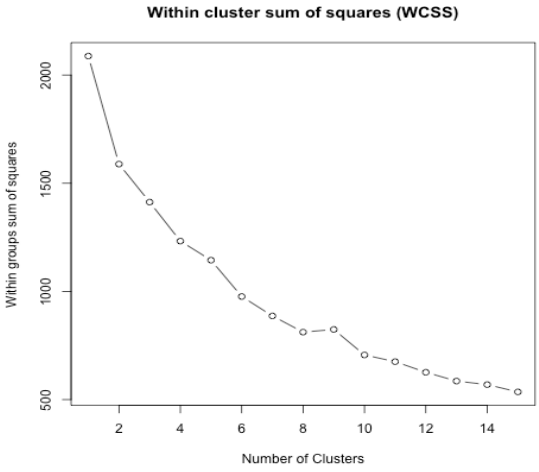
Column	Description, Indicator type
Data_Zone	2011 Data Zone, Unique code
Total_population	2014 NRS[4] small area population estimate, Count
Working_age_population_revised	Based on 2014 NRS small area population estimate, Count
Income_rate	People who are income deprived, Percentage
Employment_rate	People who are employment deprived, Percentage
CIF	People who have a limiting long-term illness or poor general health, Standardized ratio[5]
ALCOHOL	Hospital stays related to alcohol misuse, Standardized ratio
DRUG	Hospital stays related to drug misuse, standardized ratio
SMR	Standardized mortality ratio, Standardized ratio
DEPRESS	Population being prescribed drugs for anxiety, depression or psychosis, Percentage
LBWT	Live singleton births of low birth weight, Percentage
EMERG	Emergency stays in hospital, standardized ratio
Attendance	School pupil attendance, Percentage
Attainment	School leavers who attaining a qualification, Score
Noquals	Working age people with no qualifications, Standard ratio
NEET	People aged 16-19 not in full time education, employment or training, Percentage
HESA	17-21year olds entering in to full time higher education, Percentage
drive_petrol	Average drive time to a petrol station, Time (minutes)
drive_GP	Average drive time to a GP surgery, Time (minutes)
drive_PO	Average drive time to a post office, Time (minutes)
drive_primary	Average drive time to a primary school, Time (minutes)
drive_retail	Average drive time to a retail center, Time (minutes)
drive_secondary	Average drive time to a secondary school, Time (minutes)
PT_GP	Public transport travel time to a GP surgery, Time (minutes)
PT_Post	Public transport travel time to a post office, Time (minutes)
PT_retail	Public transport travel time to a retail center, Time (minutes)
crime_rate	Recorded crimes of violence, Count per 10,000 population
overcrowded_rate	People in households that are overcrowded, Percentage
nocentralheat_rate	People in households without central heating, Percentage

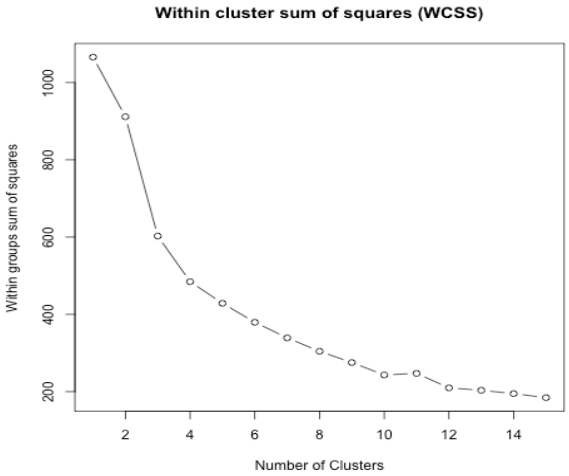
[4] National Records of Scotland

[5] A value of 100 is the Scotland average, values greater than 100 indicates poorer condition relative to Scotland and vice-versa

Appendix B: WCSS plots

Figures below are WCSS plots for the first and second datasets (left and right). Generally minimizing the WCSS will maximize the distance between clusters. I selected 10 clusters beyond there is no significant decrease in WCSS beyond 10 clusters.





Appendix C: The R code

```
setwd("~/Desktop/Visual Analytics/Assignment 2/Edinburgh")
set.seed(10)

#var_mod: 1: all 28 variables, 2: 6 highest varying variables
var_mod <-1
literature_vars <- c("Income_rate","Employment_rate","ALCOHOL","DEPRESS","LBWT","crime_rate")

library(kohonen)
library(ggplot2)
library(rgdal)
library(gridExtra)
library(grid)

#read in the census data for the Edinburgh area
data <- read.table(file="Edited_Data.csv", sep=",", header=TRUE)

#read in the boundary data for the Edinburgh area, already matched up by row with the census data
Edinburgh_map <- readOGR(dsn="SG_SIMD_2016_EDINBURGH", layer="SG_SIMD_2016_EDINBURGH")

#plot the spatial polygons data frame
```

```
plot(Edinburgh_map)

#convert the object into latitude and longitude for easier use with ggmap
Edinburgh_map <- spTransform(Edinburgh_map, CRS("+proj=longlat +ellps=WGS84 +datum=WGS84
+no_defs"))

#convert spatial polygon to dataframe including columns of spatial information
Edinburgh_fort <- fortify(Edinburgh_map, region= "DataZone")

#merge the new dataframe with the Edinburgh census data using their shared column (id)
Edinburgh_fort <- merge(Edinburgh_fort, data, by.x="id", by.y="Data_Zone")

#create plots for different variables (I'm showing just for one variable here).
ggplot(data=Edinburgh_fort, aes(x=long, y=lat, fill=Total_population, group=group)) +
  scale_fill_gradientn(colours=c("blue", "green", "red"))+
  geom_polygon(colour="black")+
  coord_equal()+
  theme()
ggsave("Adjustment/jpg/ed_map_vars/Total_population.png")

#choose the variables with which to train the SOM by subsetting the dataframe called data
data_train <- data[, c(4,5,6,8,10,11,12,13,14,15,16,17,18,19,20,21,22,23,24,25,26,27,28,29,30,32,35,36)]
#standardise the data creating z-scores and convert to a matrix
data_train_matrix1 <- as.matrix(scale(data_train))
#keep the column names of data_train as names in our new matrix
names(data_train_matrix1) <- names(data_train)

if (var_mod==1){
  data_train_matrix = data_train_matrix1
} else{# variables  with highest variance
  X = data_train
  #Normalize
  X = (X - min(X))/(max(X) - min(X))
  #Compute variances
  vars = numeric(ncol(X))
  for (i in 1:ncol(X)){
    vars[i] = var(X[,i])
  }
  varsIdx = order(vars,decreasing=TRUE)
  selected_vars = varsIdx[1:10]
  data_train_matrix = data_train_matrix1[,selected_vars]
}

#define the size and topology of the som grid
som_grid <- somgrid(xdim=9, ydim=9, topo="hexagonal")

# Train the SOM model!
som_model <- som(data_train_matrix,
                 grid=som_grid,
```

```
            rlen=1000,
            alpha=c(0.1,0.01),
            keep.data = TRUE )

# Plot of the training progress - how the node distances have stabilised over time.
#mean distance to closes codebook vector during training
plot(som_model, type = "changes")

mydata <- getCodes(som_model)
wss <- (nrow(mydata)-1)*sum(apply(mydata,2,var))
for (i in 2:15) wss[i] <- sum(kmeans(mydata,
                             centers=i)$withinss)

png(filename=paste("Adjustment/jpg/wss",var_mod,".png", sep=""))
plot(1:15, wss, type="b", xlab="Number of Clusters",
    ylab="Within groups sum of squares", main="Within cluster sum of squares (WCSS)")
dev.off()

# Form clusters on grid
# use hierarchical clustering to cluster the codebook vectors
som_cluster <- cutree(hclust(dist(getCodes(som_model))), 10)

# Colour palette definition
pretty_palette <- c("#1f77b4", '#ff7f0e', '#2ca02c', '#d62728', '#9467bd', '#8c564b', '#e377c2', '#ffffff', '#ffff00',
'#660066' )

#show the same plot with the codes instead of just colours
png(filename=paste("Adjustment/jpg/codes",var_mod,".png", sep=""))
plot(som_model, type="codes", bgcol = pretty_palette[som_cluster], main = "Clusters")
add.cluster.boundaries(som_model, som_cluster)
dev.off()

cluster_details <- data.frame(id=data$Data_Zone, cluster=som_cluster[som_model$unit.classif])
mappoints <- merge(Edinburgh_fort, cluster_details, by="id")

ggplot(data=mappoints, aes(x=long, y=lat, group=group, fill=factor(cluster))) +
  geom_polygon(colour="transparent")  +
  coord_equal() +
  scale_fill_manual(values = pretty_palette)

ggsave(paste("Adjustment/jpg/ed_map/",var_mod,".png",sep=""))

Edinburgh_map <- merge(Edinburgh_map, data, by.x="DataZone", by.y="Data_Zone")
Edinburgh_map <- merge(Edinburgh_map, cluster_details, by.x="DataZone", by.y="id")

#saving heatmaps
for (var in 1:ncol(data_train_matrix)){
  png(filename=paste("Adjustment/jpg/heatmaps",var_mod,"/", names(data_train_matrix[1,])[var],".png", sep=""))
```

```
  plot(som_model, type = "property", property = getCodes(som_model)[,var],
main=colnames(getCodes(som_model))[var], palette.name=coolBlueHotRed)
  dev.off()
}

#geographical maps
if (var_mod==1){
  #Now, plot together the first 16

  png("Adjustment/heatmaps16_1.png")
  par(mfrow=c(4,4),tcl=-0.5,mai=c(0.5, 0.5, 0.5, 0.5),mgp = c(1.5, 0.5, 0))
  for (var in 1:16){
    plot(som_model, type = "property", property = getCodes(som_model)[,var],   main =
colnames(getCodes(som_model))[var], palette.name=coolBlueHotRed)
  }
  dev.off()

  png("Adjustment/heatmaps16_2.png")
  par(mfrow=c(4,4),tcl=-0.5,mai=c(0.5, 0.5, 0.5, 0.5),mgp = c(1.5, 0.5, 0))
  for (var in 17:28){
    plot(som_model, type = "property", property = getCodes(som_model)[,var],      main =
colnames(getCodes(som_model))[var], palette.name=coolBlueHotRed)
  }
  dev.off()
}

#create a label set for our SOM - a random subset of EDNAME
#extract the the names of areas in Edinburgh
geog_names <- Edinburgh_map@data$Intermedia
#most EDNAME values are repeated, so we'll remove duplicates, just to get an idea of the spread across Edinburgh
(although you may be interested in how different areas under the same name differ)
geog_names[duplicated(geog_names)] <- NA
#find the index of the names which are not NA
naset <- which(!is.na(geog_names))
#make all but 10 of the placenames in our data NA
naset <- sample(naset, length(naset)-20)
geog_names[naset] <- NA

# Replot our data with labels=geog_names
png(filename=paste("Adjustment/jpg/clusters",var_mod,".png", sep=""))
plot(som_model, type="mapping", bgcol = pretty_palette[som_cluster], main = "Clusters", labels=geog_names)
add.cluster.boundaries(som_model, som_cluster)
dev.off()

# load custom palette, created by Shane Lynn
source('coolBlueHotRed.R')

plot(som_model, type = "counts", main="Node Counts", palette.name=coolBlueHotRed)
plot(som_model, type = "quality", main="Node Quality/Distance", palette.name=coolBlueHotRed)
```

```
png(filename=paste("Adjustment/jpg/Umat",var_mod,".png", sep=""))

plot(som_model, type="dist.neighbours", main = "SOM neighbour distances", palette.name=grey.colors)
add.cluster.boundaries(som_model, som_cluster)
dev.off()

plot(som_model, type = "codes")

#write our Edinburgh_map as an esri shapefile
writeOGR(obj=Edinburgh_map, dsn="Edinburgh_map_clustered", layer="Edinburgh_map_clustered",
driver="ESRI Shapefile")
```

References:

[1] P. Townsend, Deprivation, Journal of social policy 16(2) (1987) 125-146.
[2] P. Townsend, P. Phillimore, A. Beattie, Health and deprivation: inequality and the North, Routledge1988.
[3] P. Townsend, Poverty in the United Kingdom: a survey of household resources and standards of living, Univ of California Press1979.
[4] S. Lynn, Self-Organising Maps for Customer Segmentation using R, Slideshare, 2014.
[5] S. Lynn, Self-Organising Maps for Customer Segmentation, 2014.
[6] I.K. Center, Kohonen Node, 2018. https://www.ibm.com/support/knowledgecenter/en/SS3RA7_15.0.0/com.ibm.spss.modeler.help/kohonennode_general.htm.
[7] J. Frost, How to Identify the Most Important Predictor Variables in Regression Models | Minitab, (2016).
[8] R. Wehrens, Package kohonen, 2017. https://cran.r-project.org/web/packages/kohonen/index.html.
[9] D.M. Fergusson, J.M. Boden, L.J. Horwood, Tests of causal links between alcohol abuse or dependence and major depression, Archives of general psychiatry 66(3) (2009) 260-266.
[10] S. Raphael, R. Winter-Ebmer, Identifying the effect of unemployment on crime, The Journal of Law and Economics 44(1) (2001) 259-283.
[11] I. Kawachi, B.P. Kennedy, R.G. Wilkinson, Crime: social disorganization and relative deprivation, Social science & medicine 48(6) (1999) 719-731.