Lab03: Regular Hierarchical Clustering Under Spatial Constraints

Date: Friday, March 28, 2025

Introduction

In this lab, we are going to explore possible clusters that can be constructed from data derived from all 254 counties of the state of Texas. This unsupervised machine learning approach is simply meant to examine the possibility of parsing the state into differentiated geospatial regions from the lens of policy analysis and production. As such, a search for sociodemographic variables will predominate.

The goal of this analysis is to be able to suggest some socioeconomic difference between regions of Texas that may be relevant from a policy-proposal point of view. Specifically, however, it is hoped to find sociodemographic groupings that are spatially cohesive. Therefore, hierarchical spatial clustering analysis will be used as the primary tool of exploration in this lab. All code used is reproduced at the end of this report for reference.

Feature Selection

Because of complications in interpreting distanced measures derived from non-numeric variables, I removed all 'factor' variables from the feature data-frame after having extracted it from the 'county.shp' file. I also removed identification and coordinate variables, as I only wanted to focus on broadly socio-economic attributes of the counties in Texas. This amounted to removing the following 13 variables from the feature data frame:

LONG,LAT,SEQID,FIPS,NAME_x,REGION,LINKNAME,COUNTY,NAME_y,URBRURAL,TXDOTREGIO,RELIGDOMI

Next, using the 'purrr' and 'caret' packages, I make a copy of the feature data-frame wherein every attribute was turned into a 'numeric' variable, and off of that data-frame built a correlation matrix between all the remaining variables. I next sub-setted this feature data-frame to find all variable pairings that had a correlation coefficient of 0.85 or above. I then removed these variables from the feature data-frame. This resulted in a pared down feature data-frame with 26 variables.

The reason I did this was to remove features that were too highly correlated. Such highly correlated variables may 'overstate' their influence when calculating the distance between their features. Therefore, I opted to remove them when doing this hierarchical cluster analysis, as it is based upon such distance measurements.

From the remaining features, I selected a small number based off their conceptual 'candidacy' to elucidate some socioeconomic conditions of the counties in the dataset. These 8 variables were those chosen:

PARTBLACK, SINGLEMOM, RENTVACPCT, UNEMPL, CRIMERATE

PARTBLACK is the proportion of the population that is African-American in the county, SINGLEMOM is the proportion of single mother families, RENTVACPCT is the percent of vacant rental housing, and UNEMPL is the proportion of the county population that is currently unemployed, CRIMERATE is the numbers of crimes committed per 100,000 people.

Each of these variables is correlated with the socioeconomic status of population strata and can serve as a proxy for the relative standing of a region largely composed of those strata. The proportion of black residents, single-mothers, the rate per subpopulation at which crimes are committed, and those who are unemployed are all salient features of under-served populaces. However, the composition of housing attributes can also reflect the living conditions, and hence abilities to pay for given levels of living, of certain population segments. All this is to say that each of these attributes can factor as a good stand in for socioeconomic status in certain conditions.

These variables are already in rate form, but I proceeded to 'scale' the feature data-frame. This is because 'scaling' brings each column with the range from 0 to 1. Standardization, however, involves setting each variable's standard deviation to 1, and this process is handled internally via the 'hclustgeo' package.

Looking at a correlation plot of all the chosen features:

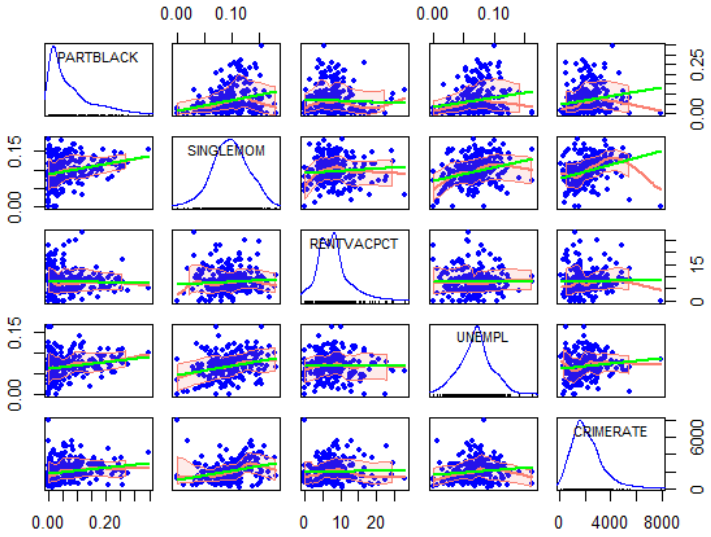

Figure 1

It can be seen that while some variables, such as SINGLEMOM and UNEMPL, do trend together, the fact that highly correlated variables have already been removed gives confidence that these features are not too redundant via one another to perform subsequent analysis on.

Spatial Matrices

I next extracted neighboring links from the counties shapefile and made a binary matrix from them. I then converted that contiguity matrix into a spatial dissimilarity matrix. This allowed me to make a map of counties within the state of Texas, as reproduced below:

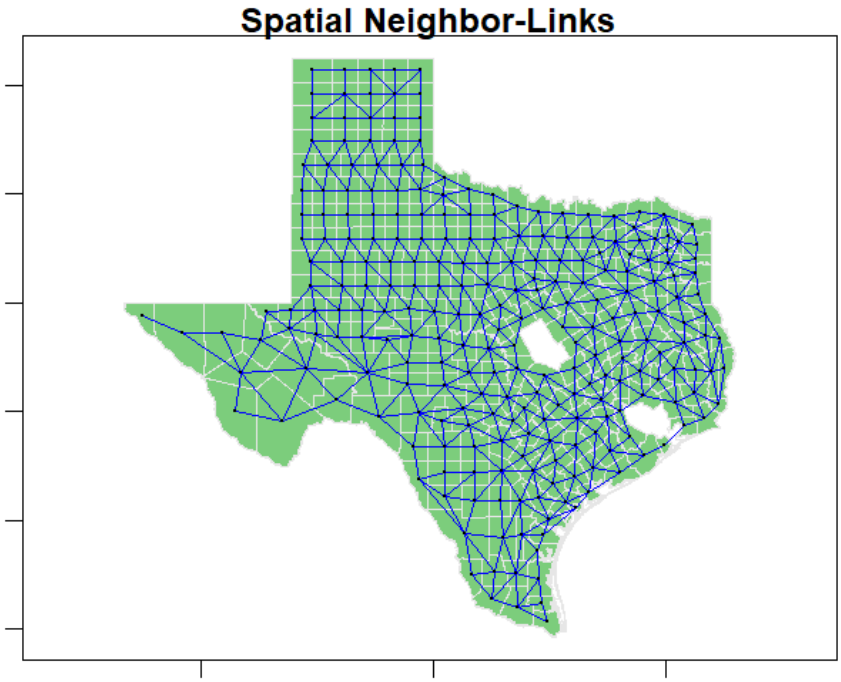

Figure 2

Cluster Count

I chose to select my clusters such that each cluster had at least more than five counties within it. However, a few counties consistently gave me trouble when picking various possible baskets of variable combinations. While this does not mean that each of these counties was a problem for every possible cluster/feature combination, the frequency with which they often dominated the lower *county-per-cluster* count led me to categorize them as outliers. I therefore removed the following counties from my 'counties.ship' file:

Harris, Coryell, and Bell Counties

I also looked at a dendrogram in order to help visually select the number of clusters that I should partition the counties into. The dendrogram shows the aggregation decisions made for groups of counties. One should only pay attention to the vertical dimension, which marks the similarity between

clusters, and not try to read closeness as being mapped in any way along the horizontal axis. As one moves up the vertical axis, each grouping that can be 'perforated' from that point is more similar.

One can select the partition of the data at a certain 'cut-point' along the vertical axis. One can choose such a point so as to divide the graph at the groupings that one thinks neatly divides the data clusters that seem to group together with each other more naturally than with other groupings. I chose a 'cut-point' of 0.39, which resulted in five cluster groupings. See 'Figure 3' below.

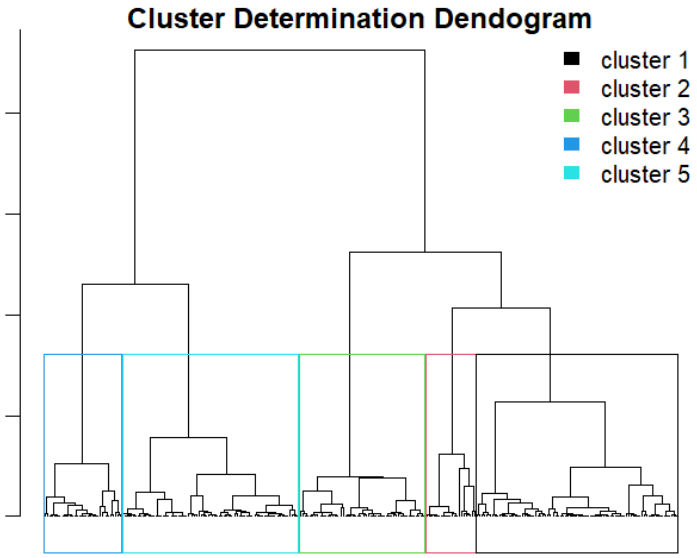

Figure 3

Using the 'tibble' and 'dplyr' packages, I also made a scree-plot from the dendrogram heights to try to mark the optimal number of clusters using the 'elbow-rule'. In this case, this rule advises that one choose the number of clusters at the point where major reductions in similarity (height) no longer occur. While there is no extra-prominent such point (see 'Figure 4'), it still suggested the selection of 5 clusters to capture a reasonable partition of the county groupings.

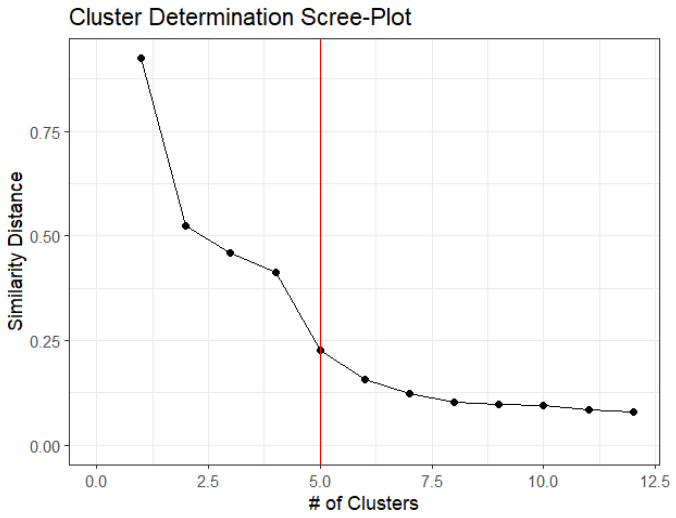

Figure 4

α, Trade-offs, and Verification

In order to perform spatially constrained hierarchical cluster analysis using the 'hclust' package, one must select a number from 0 to 1 for hyperparameter 'alpha' that determines how much geographical homogeneity is maintained in the clustering process. A higher value of alpha enforces more geographic cohesion among the clusters.

Thankfully, one can plot a range of possible alpha values given a choice of clusters. Specifically, one can plot the proportion of inertia (basically how well a data-set is partitioned by a given number of clusters) explained either by the feature or geographic distance matrix, and compare them to each for a given cluster partition. Usually, it is recommended to select an alpha value where the explanatory 'power' of the two matrices overlap.

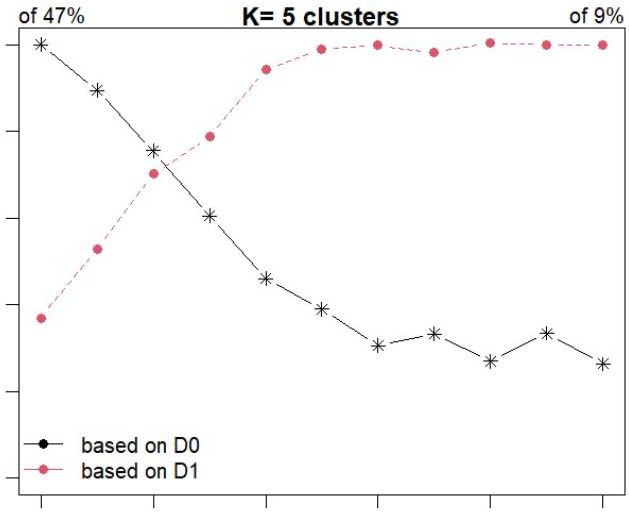

Figure 5

One way of looking at that overlap can be seen in 'Figures 5' above. From the graph, it appears 0.2 represents a reasonable alpha value, as both the feature matrix and the geospatial dissimilarity matrix roughly account for the same proportion of the data for a cluster choice of 5 at that value.

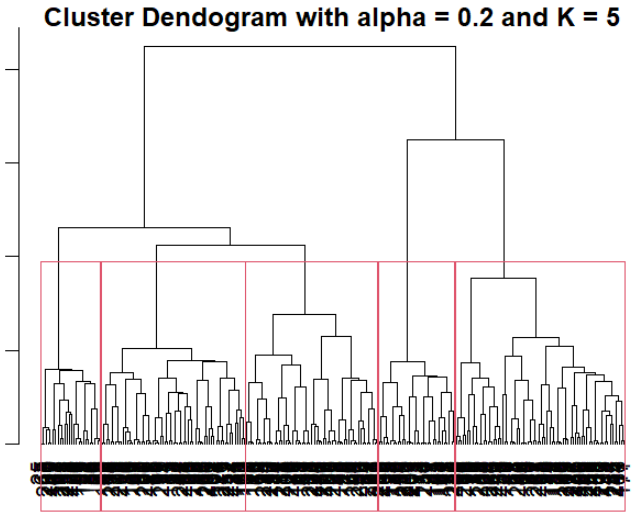

Figure 6

The outcome of performing hierarchal cluster analysis given the choice of hyperparameters K and alpha appears in 'Figure 6' above. While the x-axis labels are highly compressed due to the number of counties under consideration (251), the number of counties assigned to each cluster is as follows:

```
countyClus
 1  2  3  4  5
33 62 73 26 57
```

Figure 7

Note this distribution meets the threshold I set out with (more than five counties per cluster). Counties are also fairly-well apportioned across clusters. This served as further validation of my choice of K and alpha.

'Figure 6' shows us the 'height' or the value at which clusters are merged using Ward's linkage method. This is the red horizontal 'cut-point line' that tops each of the five clusters in the data-set. As one moves vertically up the dendrogram, the more similar would be the clusters chosen at any given cut-point. If one chooses a cut-point lower down on the vertical-axis, the partitions chosen would be even more dissimilar to each other than those that are currently selected.

While perhaps slightly unbalanced, the clusters produce a geographic spread that reflects broad contiguity ('Figure 8').

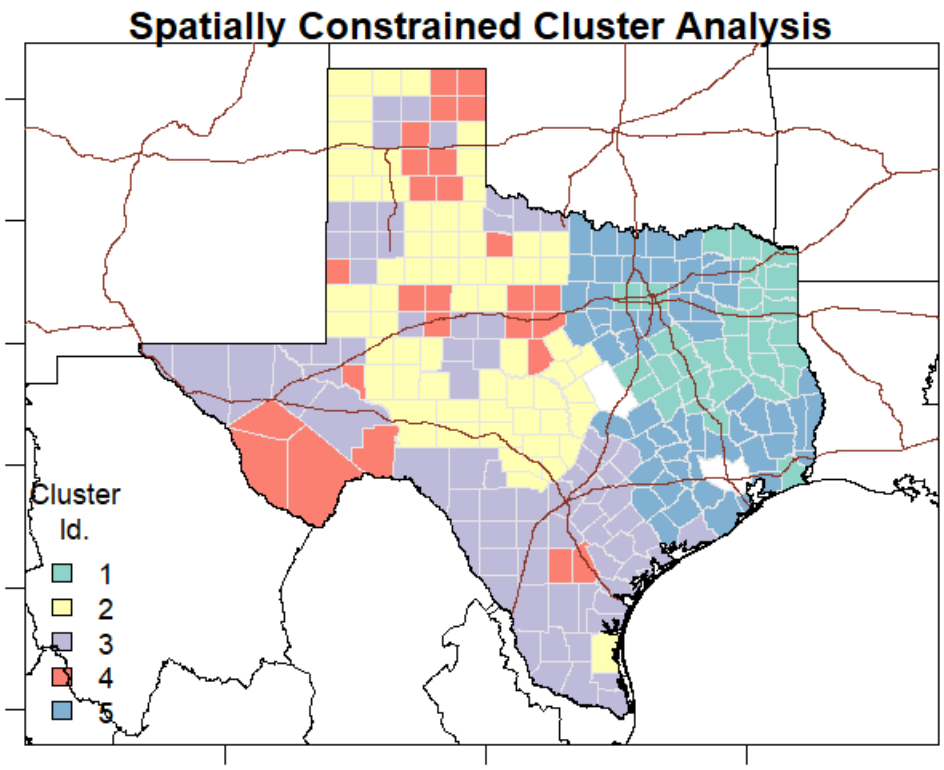

Figure 8

In order to verify the choice of clusters, we can look at the distribution of them as regards the pared down feature set. A boxplot is a good way to do this. In 'Figure 9', it can be seen that cluster one is noticeable for its share of PARTBLACK, cluster two for its low share of RENTVACPCT and CRIMERATE, cluster three for a high share of UNEMPL, cluster four for its high share of RENTVACPCT and low UNEMPL, and cluster five for its simultaneous low share of SINGLEMOM and high CRIMERATE.

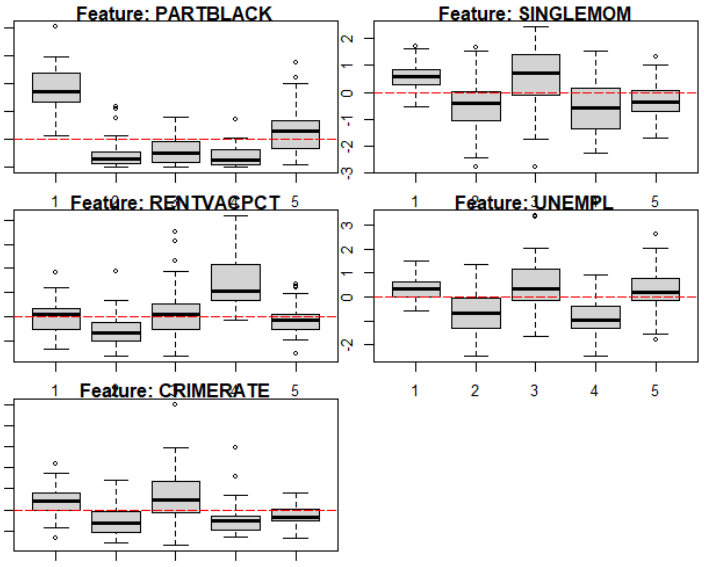

Figure 9

Not all of these clusters are completely homogenous. Cluster five suffers the most from not standing out in terms of a singular feature (but still unique in its combined ranking of two features as noted above). But the resulting geographic spread is informative. Most prominently clusters three, two, and five span over separate parts of Texas smoothly. However, overall, the analyst chose to prioritize cluster homogeneity over spatial contiguity. For example, the spread of cluster four makes more sense in its feature spread than its geographic extent. Indeed, experimenting with various values of alpha and K produced conglomerations that either did not distinguish themselves in terms of their feature space and/or resulted in clusters too unconnected geographically.

This choice of clustering shows us that the state of Texas *can* be seen to vary in two ways geospatially: by state region and along its north-south axis. Perhaps most interesting is this clustering suggests that the usual proxies of socioeconomic status vary in indirect ways.

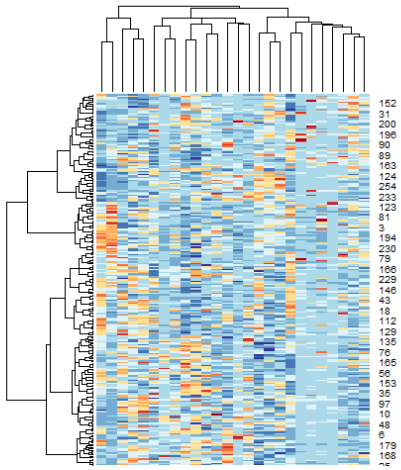

Figure 10

'Figure 10' displays a heatmap comparing features and geographic similarities between counties. In the bottom-left quarter we can see a congregation of highly correlated locations and attributes for some counties. Although it is true that the assignment of every county does not reveal an overbearing positive correlation between its geographic cluster and its feature cluster. This in in alignment with the choice to focus on attributes among clusters noted above.

However, good geo-consilience of clustering can be found when comparing the results with variables not used in producing those results. Below, it can be seen there is some broad overlap between the clusters, religious domination, and the regions of Texas. In the case of religion, the overlap is pertinent amongst heavily catholic regions. While not directly supporting the division found via the clustering above, it does indicate the possibility of the clusters correlating with a wider range of potential demographic or socioeconomic variables.

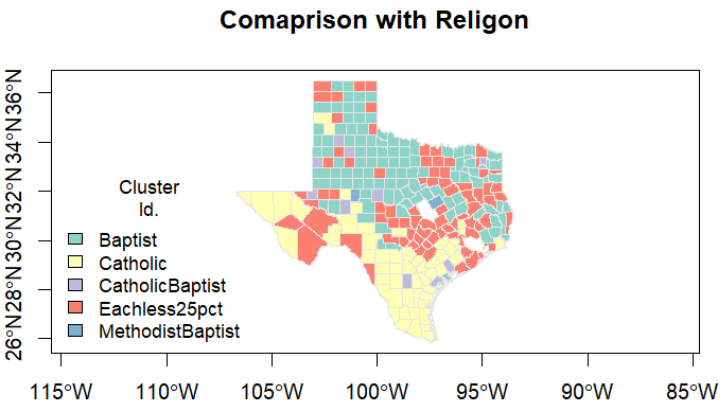

Figure 11

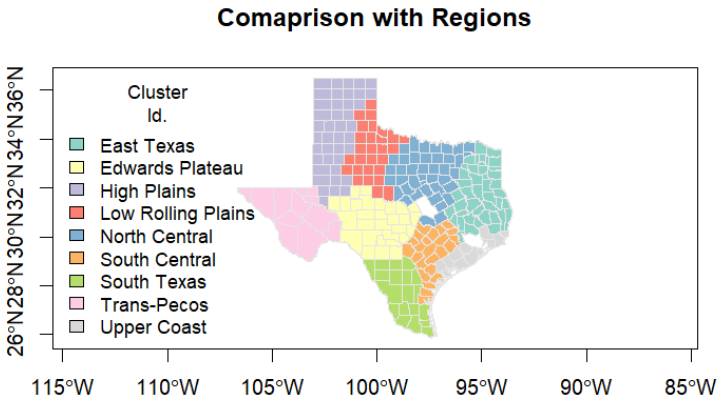

Figure 12

Interpretation and Conclusion

Most pertinent is what patterns unfold from this analysis of underlying similarity. If I were to take the perspective of a policy-analyst I would probably emphasize the need to bring income assistance to cluster five to combat low income among single-mother families. I would consider employment assistance programs, such as job retraining, to cluster three for the same reason. Cluster four could benefit from housing assistance increases. Cluster one may benefit from family-intervention or community-to-family based programs. Such as foodbanks and neighborhood crime-stopping programs.

While it would be advised to take other measurements into account (at the very least) before allocating public funds in such a direct manner, it is interesting that the clusters as construed give an indication of the differences of the populations residing throughout each of them. While no doubt each cluster could benefit from a range of social programs, it is still the case that some *might* be in more need of certain kinds of help than others. This could assist policy-work in the future to most efficiently send resources

where it needs to go to 'do the most good' as it were. Simply being able to consider such possibilities was the goal of such analysis here.

R Code Appendix

Code in the beginning is the same as 'stater code', with the following exception added after importing the 'counties.shp' file:

```
# Remove out-lier counties

# Always only had one k and multiple features, so not distinguished

county.shp <- county.shp[county.shp$NAME_x!="Harris", ]

county.shp <- county.shp[county.shp$NAME_x!="Coryell", ]

county.shp <- county.shp[county.shp$NAME_x!="Bell", ]
```

After starter code section:

```
# We only want quantitative vars. for our feature analysis.

# Remove co-ordinate variables because co-ordinates are not attributes
here.

df = subset(dfTX, select = -
c(LONG,LAT,SEQID,FIPS,NAME_x,REGION,LINKNAME,COUNTY,NAME_y,URBRURAL,TX
DOTREGIO,RELIGDOMI)) # Remove factor and co-ords vars.

str(df)


library(purrr)

df_numeric <- map_df(df, as.numeric) # Use purrr:map_df to convert all
vars in 'df' to 'num'

str(df_numeric)

cor_matrix <- cor(df_numeric) # Make a correlation matrix from df

cor_matrix

library(caret)

# Finds vars. that correlate more than 85% using
carret::findCorrelation

dalist <- findCorrelation(cor(cor_matrix), cutoff=0.85)

df2 <- subset(df, select = -dalist) # Remove highly correlated
variables

str(df2)
```

```
# Choose relevant socio-economic vars. from remaining vars.

selvars <- c("PARTBLACK", "SINGLEMOM", "RENTVACPCT", "UNEMPL",
"CRIMERATE")

df <- df[selvars]


# Plot scatter-plots of selected vars.

car::scatterplotMatrix(df,pch=20, regLine=list(col="green"),

                       smooth=list(span = 0.35,lty.smooth=1,

                                   col.smooth="salmon",
col.var="salmon"))


# if plot error

dev.off()

print(plot(1))


featDist <- dist(scale(df)) # Standardize features, a.k.a divide by
their s.d's

class(featDist)


## Identify spatial neighboring census tracts and process them

## Now, they will reflect their distances apart

##

nb <- spdep::poly2nb(county.shp, queen=F)              # extract first
order neighbors links

B <- spdep::nb2mat(nb, style="B")                     # convert
neighbor list to binary matrix

B[1:10,1:10]



## Transform the contiguity matrix into spatial dissimilarity
(distance) matrix
```

```
geoDist <- 1-B;         # Convert zero to ones and vice versa
diag(geoDist) <- 0


geoDist[1:10,1:10]
geoDist <- as.dist(geoDist)
class(geoDist)


## Visualize first order neighbors
# every connecting blue line is where census tracks share a common
boundry
plot(county.shp, col="palegreen3", border=grey(0.9), axes=T)
plot(nb, coords=coordinates(county.shp), pch=19, cex=0.1, col="blue",
add=T)
title("Spatial Neighbor-Links")


# Choose K
tree <- hclustgeo(featDist)
plot(tree,hang=-1, label=FALSE, xlab="", sub="", main="")
rect.hclust(tree, k=5, border=c(4, 5, 3, 2, 1))
legend("topright", legend=paste("cluster", 1:5), fill=1:5, bty="n",
border="white")
title("Cluster Determination Dendogram")


# Scree-plot to verify choice of K
# Reference: https://stackoverflow.com/questions/42334419/how-to-
generate-a-scree-plot-for-hierarchical-cluster-in-r/51223656#51223656
library(tibble)
library(dplyr)
ggplot(tree$height %>%
            as.tibble() %>%
            add_column(groups = length(tree$height):1) %>%
            rename(height=value),
```

```
        aes(x=groups, y=height)) +
    geom_point() +
    xlim(0, 12) + # Limit graph range to just k=12, which is max
allowed
    #geom_vline(xintercept = 9, col = "red") +
    geom_line() +
    geom_vline(xintercept = 5, color = "red") +
    labs(title = "Cluster Determination Scree-Plot",
         x = "# of Clusters", y = "Similarity Distance") +
    theme_bw()

## Evaluate mixture of feature and spatial dissimilarity.
## Experiment with different K and alpha values
## Hey, 'mapcolrqual' can only map upto 12 different groups...
##
K <- 5                            # Number of distinct clusters
range.alpha <- seq(0, 1, by=0.1)   # Evaluation range of mixing
parameter
cr <- choicealpha(featDist, geoDist, range.alpha, K, graph=TRUE)
cr
##
## Perform spatially constrained cluster analysis with selected alpha
##
tree <- hclustgeo(featDist, geoDist, alpha=0.2)
plot(tree, hang=-1, main="Cluster Dendogram with alpha = 0.2 and K =
5")
rect.hclust(tree, k=K)

## Number of census tracts per area
countyClus <- as.factor(cutree(tree, K))        # Determine cluster
membership
```

```
table(countyClus)                                    # number of tracts in
each cluster


## Map Results
##


mapColorQual(countyClus, county.shp,
             map.title ="Spatially constrained Cluster Analysis",
             legend.title="Cluster\nId.", legend.cex=0.9)


## Evaluate cluster characteristics
##
plotBoxesByFactor(df, countyClus, ncol=2, zTrans=T, varwidth=F)


# for 'margins too large'
par("mar")
dev.off()
par(mar=c(1,1,1,1))


# Big Map
mapColorQual(countyClus, county.shp,
             map.title ="Spatially constrained Cluster Analysis",
             legend.title="Cluster\nId.", legend.cex=0.9)
plot(interState.shp, col="tomato4", lwd=1, add=T)
plot(neig.shp, add=T)



## Describing Clusters with Heatmap
##
library("RColorBrewer")
col <- rev(colorRampPalette(brewer.pal(10, "RdYlBu"))(10))
```

```
## Calculate hclust dendrograms
hclust_rows <- as.dendrogram(tree)
hclust_cols <- as.dendrogram(hclust(dist(t(scale(df2)))))

## Draw heatmap with hclust dendrograms
heatmap(as.matrix(scale(df2)), col=col,
        Rowv = hclust_rows, Colv = hclust_cols,
        main="Texas Census Tract Profile",
        margins = c(0, 5))

## Map against other vars. as needed
mapColorQual(county.shp$REGION, county.shp,
             map.title=”Comprison with Religon”,
             legend.title=”Cluster\nId.”, legend.cex=0.9)
```