Computing Clusters

Manoradhan Murugesan

11/27/2016

1 Overview & Data Sources

Principle Component Analysis is a technique to reduce the number of dimensions in a dataset where the dimensions might be correlated. PCA is used in this assignment to reduce the number of dimensions which describe the socio economic states in Chicago census dataset. The variables chosen describe the per capita income, income levels, household size, racial characteristics and educational qualifications. It is possible that the educational qualification variables are correlated with the income variables, and PCA helps to reduce the number of dimensions by reducing such correlated variables by linear combinations to a set of uncorrelated principle components.

2 Findings

32 variables were reduced to 10 principle components using PCA. This reveals high correlation between certain variables. k-means clusters calculated had observations which were not spatially contiguous, revealing that similar observations need not necessarily be geographical neighbours. Also, k-means cluster sizes stabilize for 10 Principle Components and 25 initial assignments. The clusters obtained have the southern community areas in 1 cluster, except Hyde Park, Kenwood and Brownside. Brownside is found to have a very small population, all of African American ethinicity. Hyde Park and Kenwood cluster with the northern community areas, revealing their dissimilarity from other southern community areas that are poor. Contiguity constrained clustering techniques, which take into consideration the neighbours in the geographical space and constrain the clusters to be made of contiguous neighbours, results in 4 clusters which mark the departure from downtown to northern suburbs to southern community areas. It reveals that the relatively prosperous northern suburbs cluster stretches all the way to Ashburn and Garfield Ridge in the south.

3 PCA

```
library(rgeos)
```

```
## rgeos version: 0.3-20, (SVN revision 535)
##  GEOS runtime version: 3.4.2-CAPI-1.8.2 r3921
##  Linking to sp version: 1.2-3
##  Polygon checking: TRUE
```

```
library(spdep)
```

```
## Loading required package: sp

## Loading required package: Matrix
```

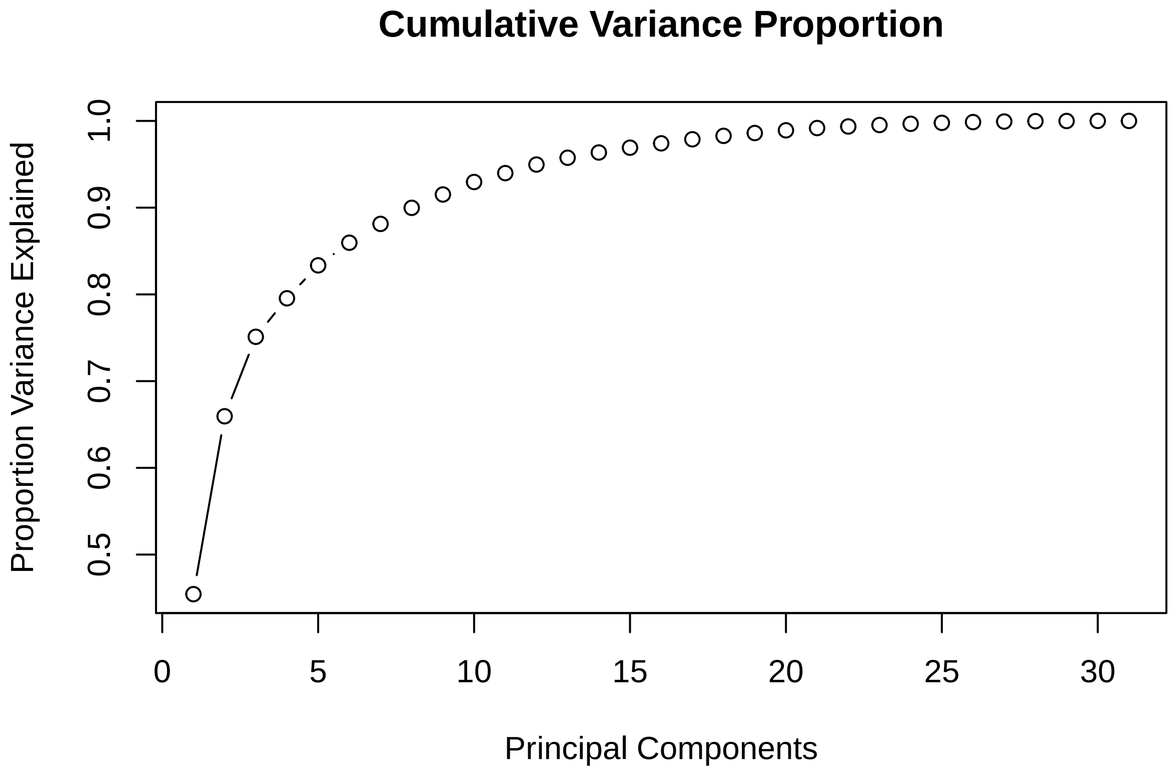
```
library(maptools)
```

```
## Checking rgeos availability: TRUE
```

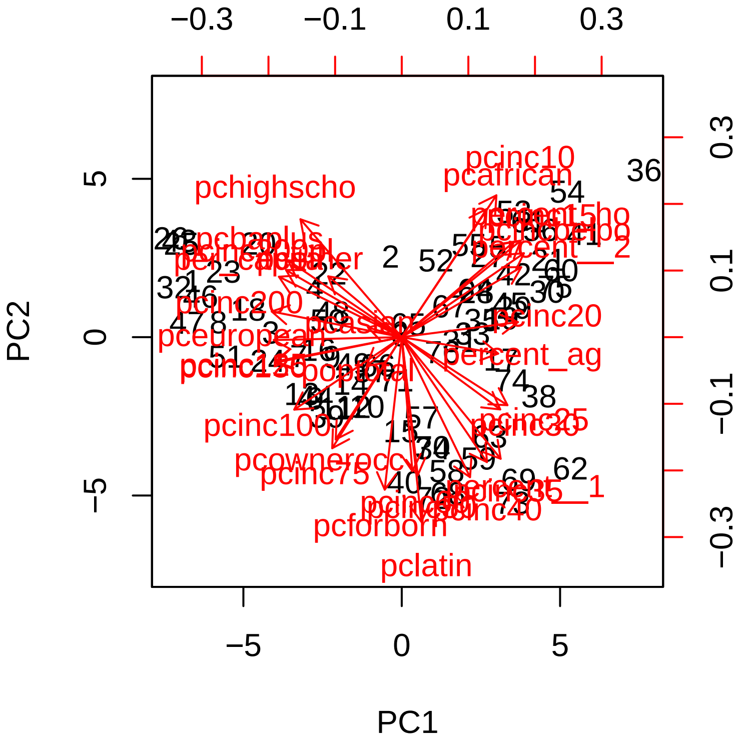
```r
library(rgdal)
```

```
## rgdal: version: 1.1-10, (SVN revision 622)
##  Geospatial Data Abstraction Library extensions to R successfully loaded
##  Loaded GDAL runtime: GDAL 1.11.4, released 2016/01/25
##  Path to GDAL shared files: /Library/Frameworks/R.framework/Versions/3.3/Resources/library/rgdal/gdal
##  Loaded PROJ.4 runtime: Rel. 4.9.1, 04 March 2015, [PJ_VERSION: 491]
##  Path to PROJ.4 shared files: /Library/Frameworks/R.framework/Versions/3.3/Resources/library/rgdal/p
##  Linking to sp version: 1.2-3
```

```r
library(RColorBrewer)
library(classInt)
library(maps)
library(ggplot2)

chicago <- readShapePoly("chicensus_merge.shp")
varnames <- c("per_capita", "percent_ho", "percent_ag", "percent__1", "percent__2",
    "pcincbelpo", "pcasian", "pcafrican", "pclatin", "pceuropean", "pcother", "pcforborn",
    "pchighscho", "pcbaplus", "pcownerocc", "pcinc10", "pcinc15", "pcinc20", "pcinc25",
    "pcinc30", "pcinc35", "pcinc40", "pcinc50", "pcinc60", "pcinc75", "pcinc100",
    "pcinc125", "pcinc150", "pcinc200", "pcinc200pl", "poptotal")


dat <- data.frame(chicago@data[, tolower(varnames)])
vd1 <- dat[, varnames]
vds <- scale(vd1)

prc <- prcomp(vds)
scree_plot <- function(princ, cumulative = FALSE) {
    pv <- princ$sdev^2
    pve <- pv/sum(pv)
    mtitle = "Scree Plot"
    if (cumulative) {
        pve <- cumsum(pve)
        mtitle = "Cumulative Variance Proportion"
    }
    plot(pve, type = "b", main = mtitle, xlab = "Principal Components", ylab = "Proportion Variance Expl
}

scree_plot(prc, cumulative = TRUE)
```



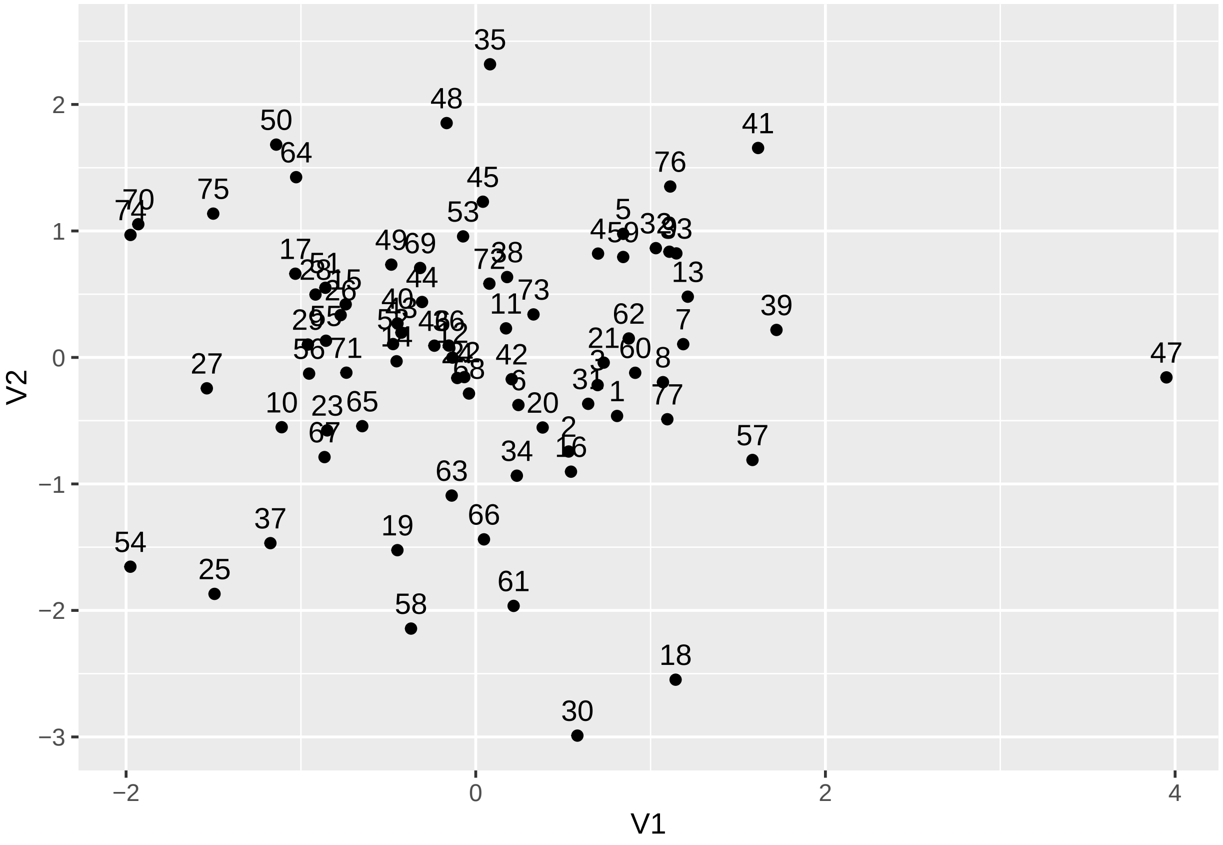
```
biplot(prc, scale = 0)
```



The scree plot reveals that more than 90% of the variance is explained by the first 10 principle components. So the 32 variables can be replaced by the first 10 principle components for further analysis.

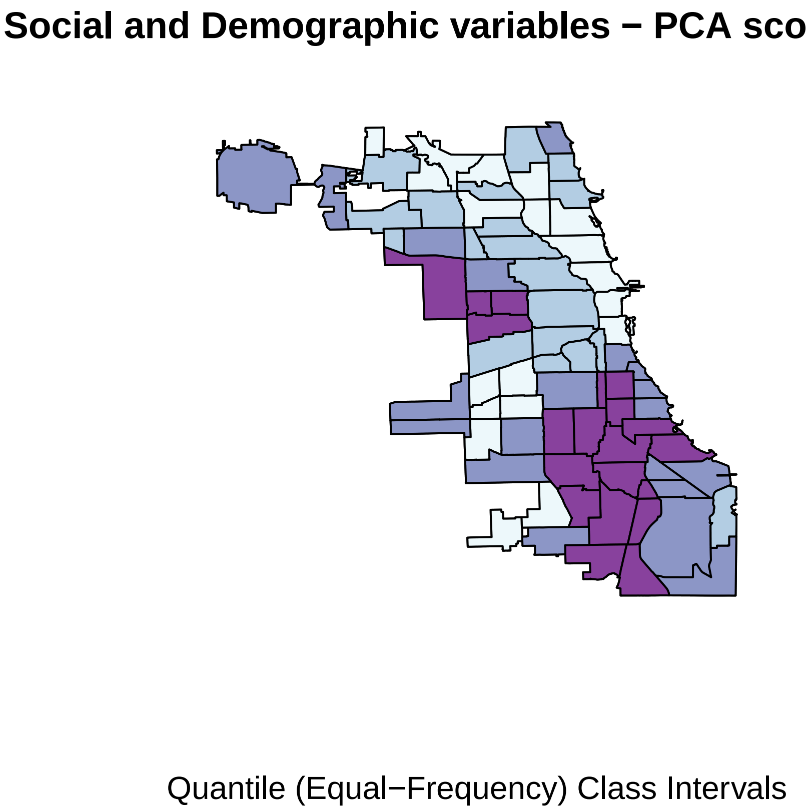
The biplot gives some information on how the variables contribute to principle components 1 and 2. It also reveals how close the variables are correlated to each other. It can be seen in the biplot that principle component 1's biggest contributed in the positive direction is 'PCINC200', the percentage of population with income levels between 150000 to 200000 USD. Its biggest contributed in the negative direction is PCINCBELPOV, the percentage of population with incomes below poverty level. For principle component 2, PCLATIN and PCINC10 contribute the most in positive and negative directions. PCLATIN is the percentage of people of Latino ethnicity and PCINC10 is the percentage of population with income levels less than 10000 USD.

```
pcscores <- prc$x
pcs1 <- as.data.frame(pcscores)
pcvarnames <- c("PC1", "PC2", "PC3", "PC4", "PC5", "PC6", "PC7", "PC8", "PC9", "PC10")

vd <- pcs1[, pcvarnames]
vds <- scale(vd)
vdiss <- dist(vds)
vmds <- cmdscale(vdiss)
datmds <- as.data.frame(vmds)
datmds$COMMAREANO <- data.frame(chicago@data)$commareano
ggplot(datmds, aes(x = V1, y = V2)) + geom_point() + geom_text(aes(label = COMMAREANO),
    nudge_y = +0.2)
```



Linking and Brushing on GeoDa with the scatter plots of PC1 and PC2 reveals that observations which are closer to each other in the scatter plot are not necessarily contiguous neighbours in the geographical space. So classical clustering techniques will not result in spatially contiguous clusters, as in the next section.

```
pcsum <- data.frame(datmds$COMMAREANO, rowSums(vd))

plotvar <- rowSums(vd)
nclr <- 4
plotclr <- brewer.pal(nclr, "BuPu")
class <- classIntervals(plotvar, nclr, style = "quantile")
colcode <- findColours(class, plotclr)
plot(chicago)
plot(chicago, col = colcode, add = T)
title(main = "Social and Demographic variables - PCA score sums", sub = "Quantile (Equal-Frequency) Clas
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```


Social and Demographic variables − PCA score sums
Quantile (Equal−Frequency) Class Intervals

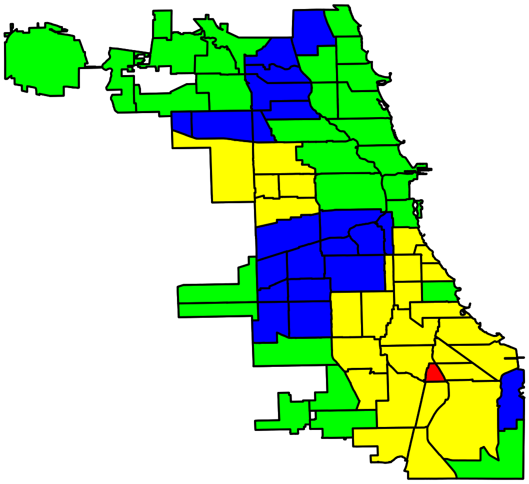
4 k-means clustering

k-means clusters are computed with 25 as the initial assignment and 1000 iterations. k-means were first computed for 20 principle components, and it was found that the cluster sizes were not stable after each run. This could be due to the higher number of dimensions, which causes the minimum and maximum distances between points to converge, resulting in k-means resolving clustering decision ties. Reducing the number of dimensions to 10 leads to an overall stable cluster size, and the clusters also tend to correspond to the expected placements, such as the richer northern community areas and poorer southern community areas.Brownside is in a cluster of size 1, and further inspection reveals that it is a community area with very small population and is polutated exclusively by people of African American ethnicity. It also is the lower outlier for many other variables such as PCINC200PLUS and PCBAPLUS.

```r
set.seed(1234567)

km1_4 <- kmeans(vds, 4, nstart = 25, iter.max = 1000)
km1_4
```

```
## K-means clustering with 4 clusters of sizes 1, 28, 21, 27
##
## Cluster means:
##          PC1        PC2          PC3         PC4         PC5        PC6
## 1  1.2222010  0.5688311  1.259091874  0.44876071  0.94858087  6.5453392
## 2 -1.0846456  0.1556053  0.004196479 -0.02663987 -0.11191321  0.1750239
## 3  0.3518583 -1.1666658 -0.593743737 -0.37954263 -0.04777891 -0.0503661
## 4  0.8058834  0.7249705  0.410815748  0.30620559  0.11808689 -0.3847526
##           PC7         PC8          PC9         PC10
## 1 -1.05913508 -0.42518774 -0.764481227  2.793270135
## 2  0.42285906  0.07568470 -0.001034651 -0.070027229
## 3 -0.44162328 -0.06745288  0.057696166 -0.009150422
## 4 -0.05580851 -0.01027679 -0.015487705 -0.023716625
##
## Clustering vector:
##  0  1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
##  2  2  4  2  2  4  2  2  2  3  3  2  3  2  3  3  2  4  2  4  2  4  2  2  2
## 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49
##  2  2  4  4  4  1  4  2  4  3  4  4  2  3  2  3  4  4  2  2  4  2  2  2  2
## 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74
##  2  2  4  4  4  4  4  3  3  3  4  4  3  3  3  4  4  4  3  3  3  3  3  3  3
## 75 76
##  4  4
##
## Within cluster sum of squares by cluster:
## [1]   0.0000 210.0671 184.4648 179.0098
##  (between_SS / total_SS =  24.5 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```r
colortab <- data.frame(cluster = km1_4$cluster)
colortab$color <- NA
colortab[colortab$cluster == 1, ]$color <- "red"
colortab[colortab$cluster == 2, ]$color <- "green"
colortab[colortab$cluster == 3, ]$color <- "blue"
colortab[colortab$cluster == 4, ]$color <- "yellow"

plot(chicago)
plot(chicago, col = colortab$color, add = T)
title(main = "Social and Demographic variables - k-means clusters", )
```

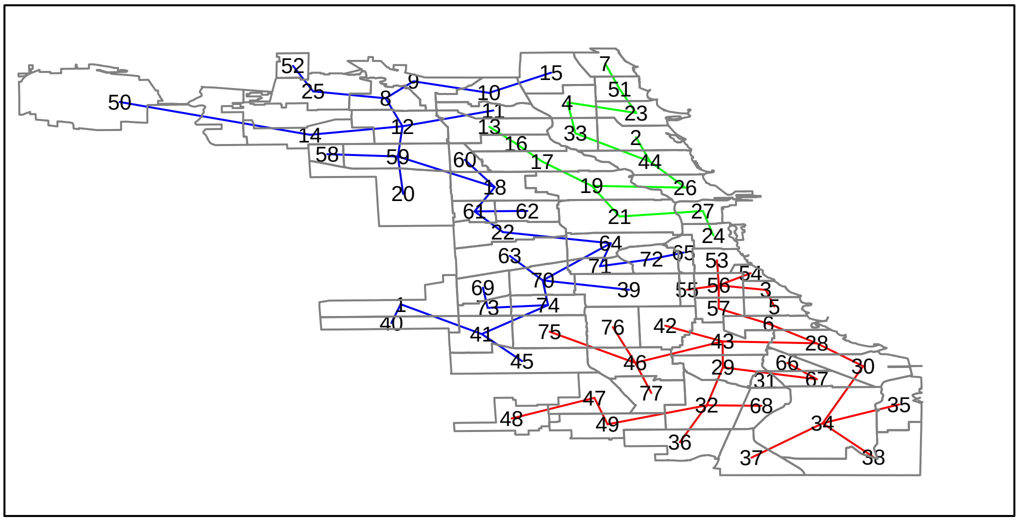


Social and Demographic variables − k−means clusters

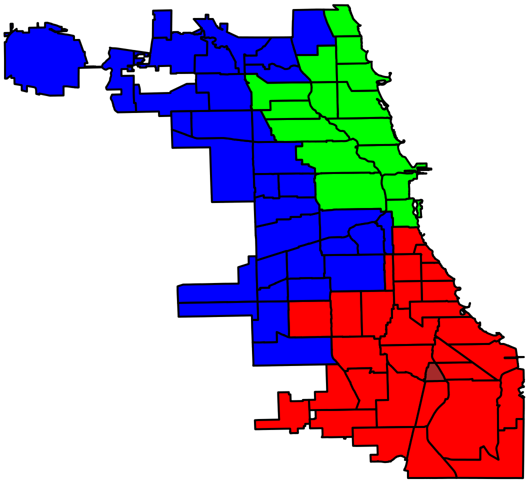
5 Contiguity constrained clustering

Constraining the clusters to be made of spatially contiguous neighbours,

```r
chicago.nb <- poly2nb(chicago)
lcosts <- nbcosts(chicago.nb, vds)
chicago.w <- nb2listw(chicago.nb, lcosts, style = "B")
chicago.mst <- mstree(chicago.w)
clus4 <- skater(chicago.mst[, 1:2], vds, 3)

plot(clus4, coordinates(chicago), cex.lab = 0.7, groups.colors = c("red", "green",
    "blue", "brown"))
plot(chicago, border = gray(0.5), add = TRUE)
```



```r
plot(chicago, col = c("red", "green", "blue", "brown")[clus4$groups])
```



Spatially constrained clusters give an overview of how Chicago's community areas' social and economic variables change from the downtown northern areas to northern suburbs and the southern community areas.