Flood Vulnerability Clusters

Manoradhan Murugesan

11/27/2016

1 Overview

Frequency of floods and potential of a region to come out of a natural disaster together decide how well a region can be impacted by floods. Flood vulnerability study of Chicago region done in this experiment identifies the contribution of social, demographical, housing and health characteristics towards resilience to floods, and, combining this with the frequency of floods for different community area, classifies Chicago community areas into a 4-point rating scale grouping of flood vulnerability.

2 Datasources

2.1 Social & demographical factors

Chicago census data, obtained from factfinder.census.gov, is aggregated to Community Area levels. the variables of interest are,

- PERCENT.HOUSEHOLDS.BELOW.POVERTY
- PERCENT.AGED.UNDER.18.OR.OVER.64
- PERCENT.AGED.16..UNEMPLOYED
- PERCENT.AGED.25..WITHOUT.HIGH.SCHOOL.DIPLOMA
- PCINCBELPOV - Percentage of population with income below poverty level
- PCASIAN - Percentage of population with Asian American ethinicity
- PCAFRICAN - Percentage of population with African American ethinicity
- PCLATIN - Percentage of population with Latino ethnicity
- PCOTHER - Percentage of popultion who cannot be clubbed under any of the above four ethnicities
- PCFORBORN - Percentage of foreign born population
- PCINC10 - Percentage of population with income of less than 10000 USD per annum

2.2 Health Characteristics

Health information sourced from Chicago Department of public health, and hosted at ChicagoHealthAtlas.org, is aggregated to Community Area levels. Variables of interest are,

- Low.Birth.Weight
- Preterm.Births
- Teen.Birth.Rate
- Assault..Homicide.
- Breast.cancer.in.females
- Cancer..All.Sites
- Colorectal.Cancer
- Diabetes.related
- Firearm.related
- Infant.Mortality.Rate
- Lung.Cancer
- Prostate.Cancer.in.Males
- Stroke..Cerebrovascular.Disease.

- Childhood.Lead.Poisoning
- Gonorrhea.in.Females
- Gonorrhea.in.Males
- Tuberculosis
- UNINSURED
- HYPERTENSION

2.3 Housing Characteristics

Chicago' Depart of Family & Support Services' 2016 HOMELESS Point-in-Time Count & Survey Report dataset, together with the American Community Survey profile data for Chicago, provide information on the housing characteristics of Chicago Community Areas. The variables of interest are,

- PCHOMELESS
- PERCENT.OF.HOUSING.CROWDED

2.4 Flooding frequency

311 call data for basement flooding and water in street flooding from 2000 through mid-September of 2016 is used to determine the frequency of floods in different community areas.

2.5 FEMA

Housing assistance data from FEMA from from 2000 through mid-September of 2016 is used as the test data to determine the validity of flood vulnerability classifications done in this study.

3 Flood vulnerability index

Flood vulnerability index is calculated using the following indicators. Their relative share to the flood vulnerability index is arrived at from the study conducted by San Francisco Department of Public Health for San Francisco.

- Social & Economic indicators - 20%
- Housing indicators - 20%
- Health indicators - 20%
- Frequency of floods - 40%

3.1 Social & Economic vulnerability index

Since 32 variables were identified from the census data, and that it is common that the variables can be correlated, PCA is used to reduce the dimensions to a smaller number of linearly uncorrelated principle components.

```r
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
```

```r
library(spdep)
```

```
## Loading required package: sp
```

```
## Loading required package: Matrix
```

```r
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/pr
##  Linking to sp version: 1.2-3
```

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

```
## Loading required package: rJava
```

```
## Loading required package: xlsxjars
```

```r
library(sqldf)
```

```
## Loading required package: gsubfn
```

```
## Loading required package: proto
```

```
## Warning in doTryCatch(return(expr), name, parentenv, handler): unable to load shared object '/Library
##   dlopen(/Library/Frameworks/R.framework/Resources/modules//R_X11.so, 6): Library not loaded: /opt/X1
##   Referenced from: /Library/Frameworks/R.framework/Resources/modules//R_X11.so
##   Reason: image not found
```

```
## Could not load tcltk.  Will use slower R code instead.
```

```
## Loading required package: RSQLite
```

```
## Loading required package: DBI
```

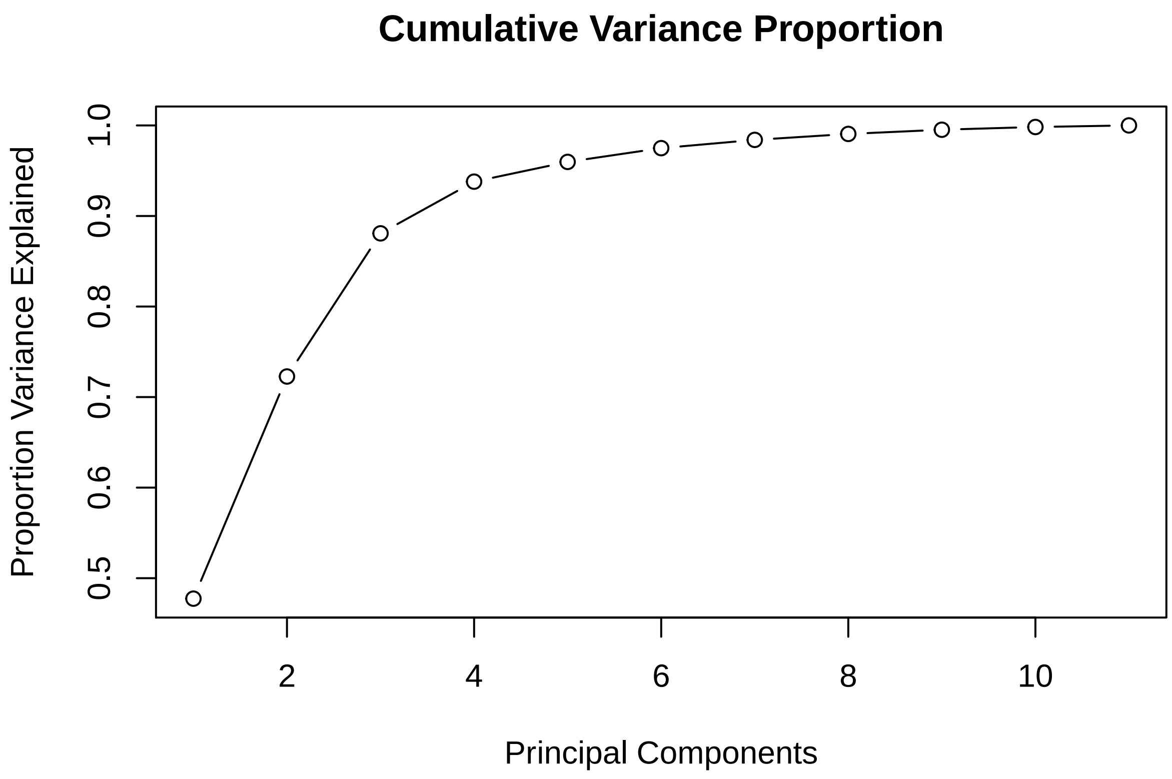
```r
library(plyr)
```

```
##
## Attaching package: 'plyr'

## The following object is masked from 'package:maps':
##
##     ozone
```

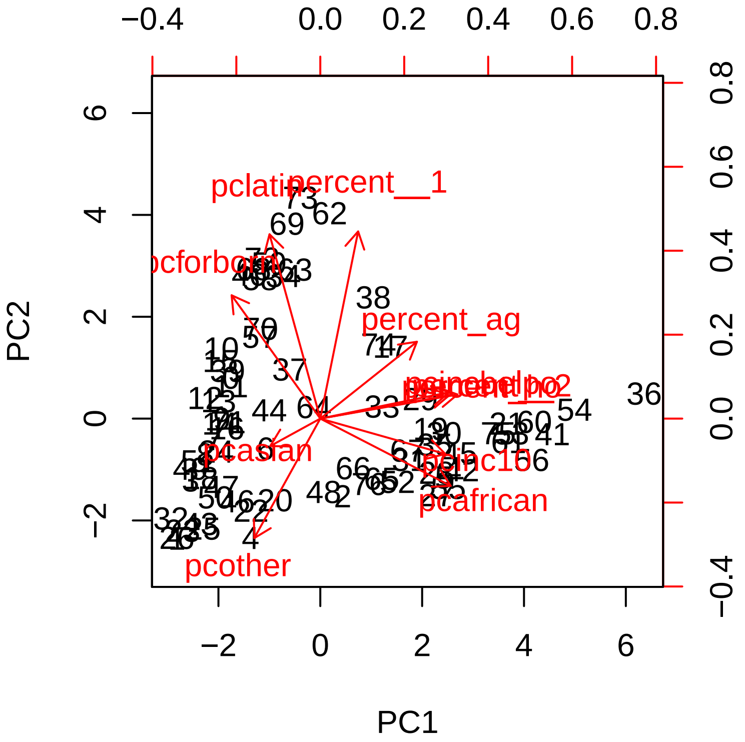
```r
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:plyr':
##
##     arrange, count, desc, failwith, id, mutate, rename, summarise,
##     summarize

## The following objects are masked from 'package:rgeos':
##
##     intersect, setdiff, union

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```r
chicago <- readShapePoly("chicensus_merge.shp")
ecovarnames <- c("percent_ho", "percent_ag", "percent__1", "percent__2", "pcincbelpo",
    "pcasian", "pcafrican", "pclatin", "pcother", "pcforborn", "pcinc10")
ecodat <- data.frame(chicago@data[, tolower(ecovarnames)])
ecovd1 <- ecodat[, ecovarnames]
ecovds <- scale(ecovd1)

ecoprc <- prcomp(ecovds)
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(ecoprc, cumulative = TRUE)
```



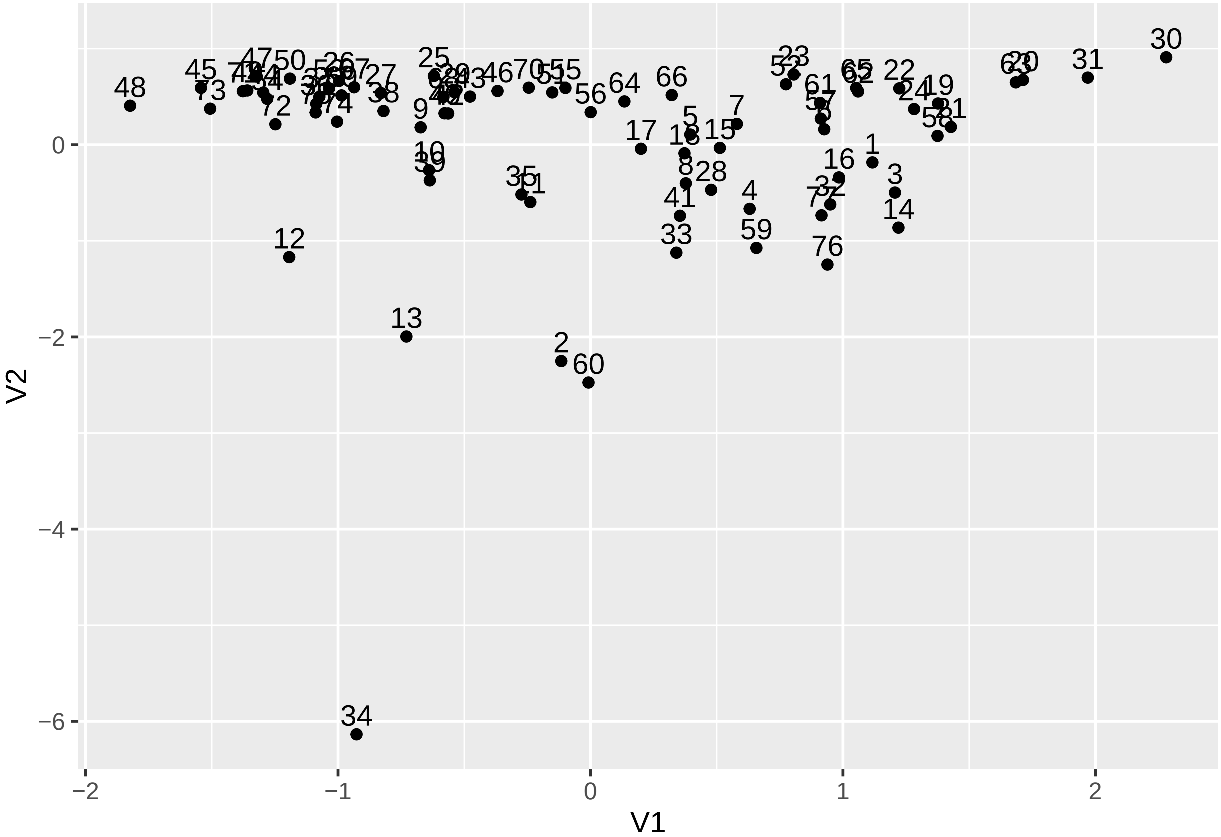
```r
biplot(ecoprc, scale = 0)
```



The scree plot reveals that more than 90% of the variance is explained by the first 4 principle components. So the 11 variables can be replaced by the first 4 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 'PCFORBORN'. Its biggest contributor in the negative direction is PERCENT.AGED.25..WITHOUT.HIGH.SCHOOL.DIPLOMA. For principle component 2, PCLATIN and PCOTHER contribute the most in positive and negative directions.

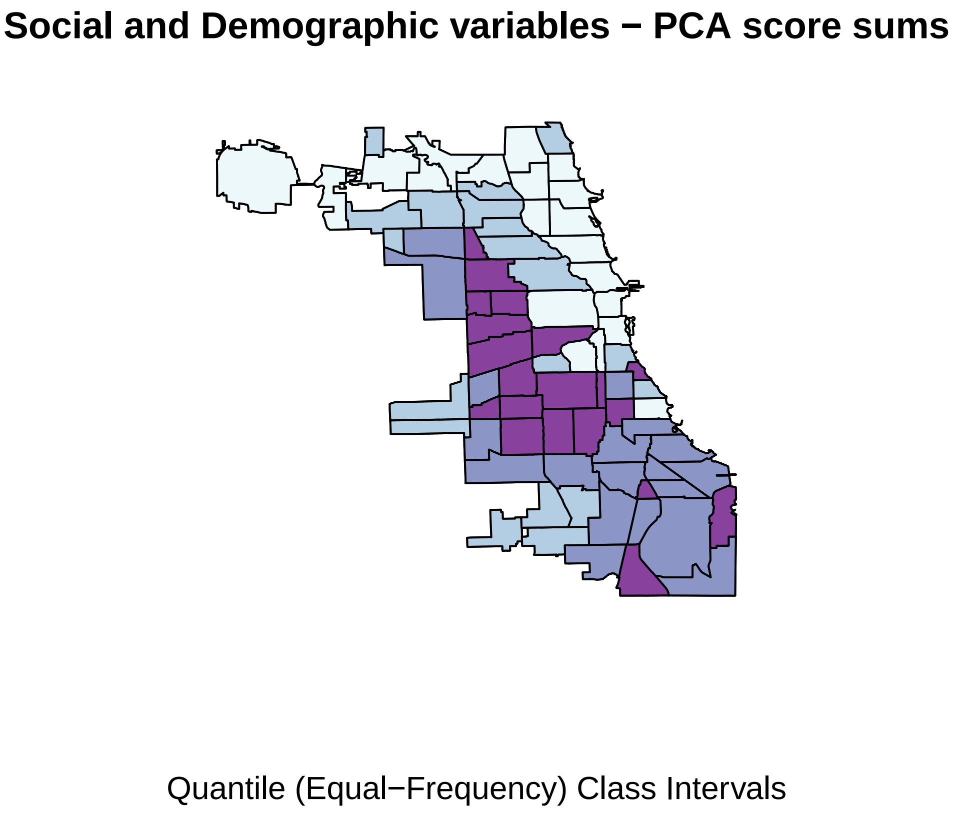
```r
ecopcscores <- ecoprc$x
ecopcs1 <- as.data.frame(ecopcscores)
ecopcvarnames <- c("PC1", "PC2", "PC3", "PC4")

ecovd <- ecopcs1[, ecopcvarnames]
ecovds <- scale(ecovd)
ecovdiss <- dist(ecovds)
ecovmds <- cmdscale(ecovdiss)
ecodatmds <- as.data.frame(ecovmds)
ecodatmds$COMMAREANO <- data.frame(chicago@data)$commareano
ggplot(ecodatmds, aes(x = V1, y = V2)) + geom_point() + geom_text(aes(label = COMMAREANO),
    nudge_y = +0.2)
```



Social vulnerability index is arrived at by adding the 4 principle components. They can be visualised in the plot below.

```r
ecopcsum <- data.frame(ecodatmds$COMMAREANO, rowSums(ecovd))
ecoIndex <- rowSums(ecovd)

plotvar <- rowSums(ecovd)
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) Cla
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

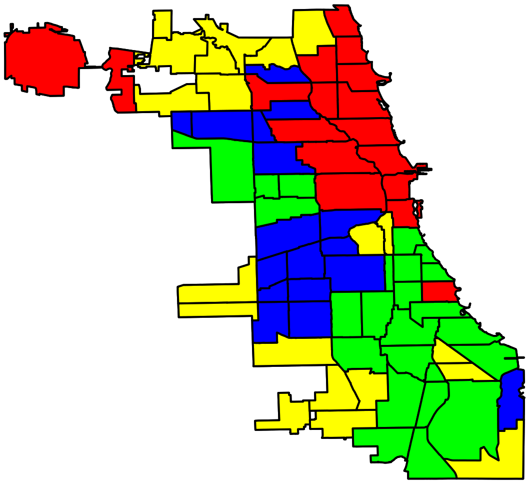
k-means clusters are computed with 25 as the initial assignment and 1000 iterations. k-means were first computed for 20 principle components, and it was found that the cluster sizes were not stable after each run. This could be due to the higher number of dimensions, which causes the minimum and maximum 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.

```
set.seed(1234567)

ecokm1_4 <- kmeans(ecovds, 4, nstart = 25, iter.max = 1000)
ecokm1_4
```

```
## K-means clustering with 4 clusters of sizes 16, 24, 17, 20
##
## Cluster means:
##           PC1         PC2          PC3          PC4
## 1 -0.9335595 -0.9184404 -0.51492245  0.99223857
## 2  1.2605155 -0.3309560  0.04245000  0.04854541
## 3 -0.2783007  1.5656575 -0.07220225  0.34589889
## 4 -0.5292155 -0.1989094  0.42236987 -1.14605941
##
## 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
##  4  1  2  1  1  2  1  4  4  4  3  4  1  4  4  3  1  3  1  2  1  2  1  1  4
## 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
##  1  1  2  2  2  2  2  1  2  3  2  2  4  3  4  3  2  2  1  4  2  4  4  4  1
## 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
##  1  4  2  2  2  2  2  3  3  3  2  2  3  3  4  4  4  2  3  3  3  4  3  3  3
## 75 76
##  2  4
##
## Within cluster sum of squares by cluster:
## [1] 13.68576 22.67434 19.76172 78.23929
##  (between_SS / total_SS =  55.8 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
colortab <- data.frame(cluster = ecokm1_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



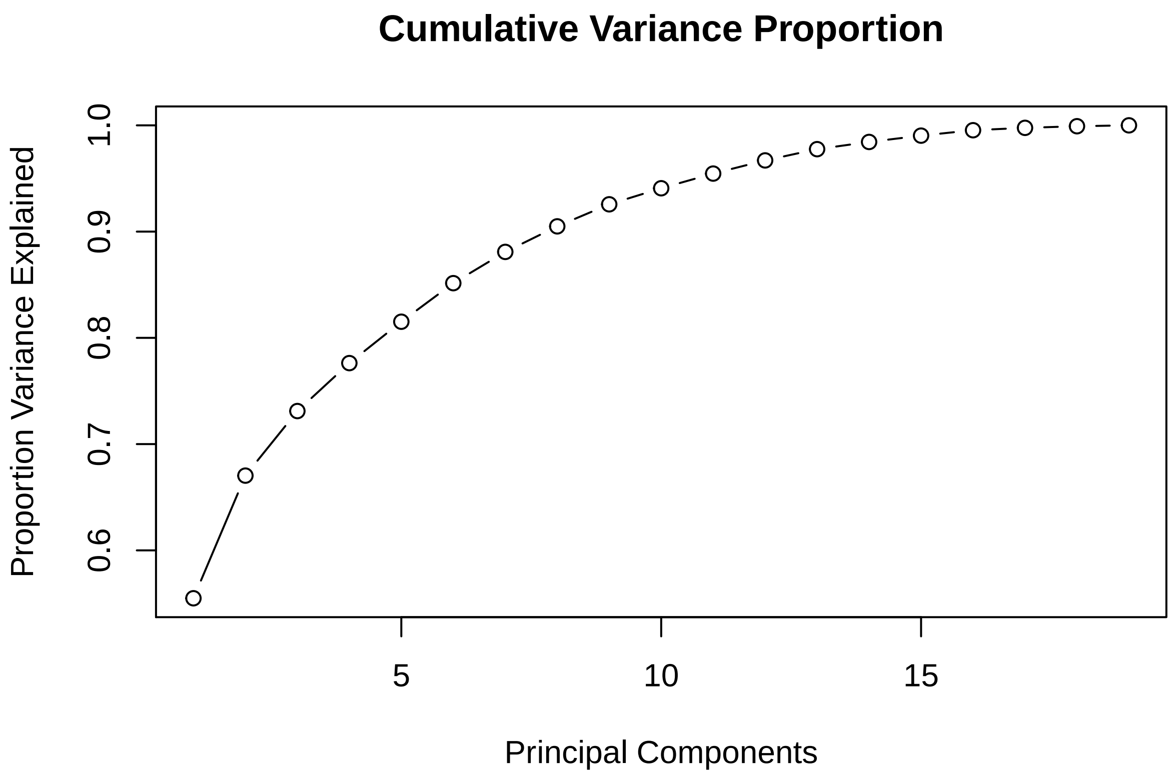
k-means clusters calculated had observations which were not spatially contiguous, revealing that similar observations need not necessarily be geographical neighbours. Also, k-means cluster sizes stabilize for 10 Principle Components and 25 initial assignments. The clusters obtained have the southern 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.

3.2 Health index

```
chihealth <- read.csv("Public_Health_Statistics-_Selected_public_health_indicators_by_Chicago_community_
chihealth[is.na(chihealth$HYPERTENSION), ]$HYPERTENSION <- 0
chihealth$Gonorrhea.in.Males <- as.numeric(as.character(chihealth$Gonorrhea.in.Males))
```

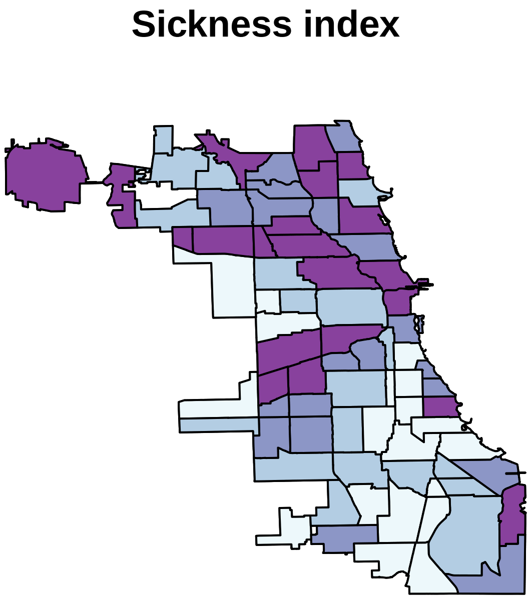
```
## Warning: NAs introduced by coercion
```

```
chihealth[is.na(chihealth$Childhood.Lead.Poisoning), ]$Childhood.Lead.Poisoning <- 0
chihealth[is.na(chihealth$Gonorrhea.in.Females), ]$Gonorrhea.in.Females <- 0

chihealth[is.na(chihealth$Gonorrhea.in.Males), ]$Gonorrhea.in.Males <- 0

hthvarnames <- c("Prenatal.Care.Beginning.in.First.Trimester", "Preterm.Births",
    "Teen.Birth.Rate", "Assault..Homicide.", "Breast.cancer.in.females", "Cancer..All.Sites.",
    "Colorectal.Cancer", "Diabetes.related", "Firearm.related", "Infant.Mortality.Rate",
    "Lung.Cancer", "Prostate.Cancer.in.Males", "Stroke..Cerebrovascular.Disease.",
    "Childhood.Lead.Poisoning", "Gonorrhea.in.Females", "Gonorrhea.in.Males", "Tuberculosis",
    "UNINSURED", "HYPERTENSION")

hthvd1 <- chihealth[, hthvarnames]
hthvds <- scale(hthvd1)

hthprc <- prcomp(hthvds)
scree_plot(hthprc, cumulative = TRUE)
```



Sickness index is arrived at by adding the 10 principle components. They can be visualised in the plot below.

```
hthpcscores <- hthprc$x
hthpcs1 <- as.data.frame(hthpcscores)
hthpcs1$COMMAREANO <- chihealth$Community.Area
hthpcvarnames <- c("PC1", "PC2", "PC3", "PC4", "PC5", "PC6", "PC7", "PC8", "PC9",
    "PC10")

hthvd <- hthpcs1[, hthpcvarnames]

hthpcsum <- data.frame(hthpcs1$COMMAREANO, rowSums(hthvd))
healthIndex <- sapply(data.frame(chicago@data)$commareano, function(x) hthpcsum[hthpcsum$hthpcs1.COMMARE
    x, ]$rowSums.hthvd.)
plotvar <- healthIndex
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 = "Sickness index", sub = "Quantile (Equal-Frequency) Class Intervals")
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```

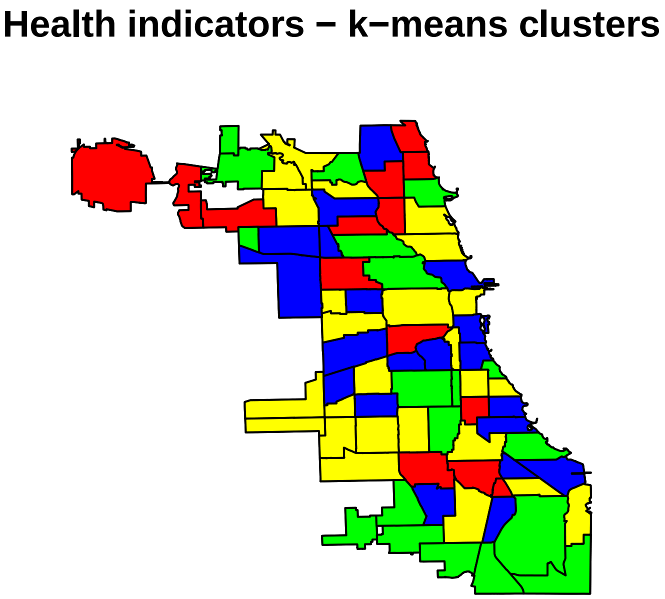


Quantile (Equal−Frequency) Class Intervals

```r
set.seed(1234567)

hthkm1_4 <- kmeans(hthvds, 4, nstart = 25, iter.max = 1000)
hthkm1_4
```

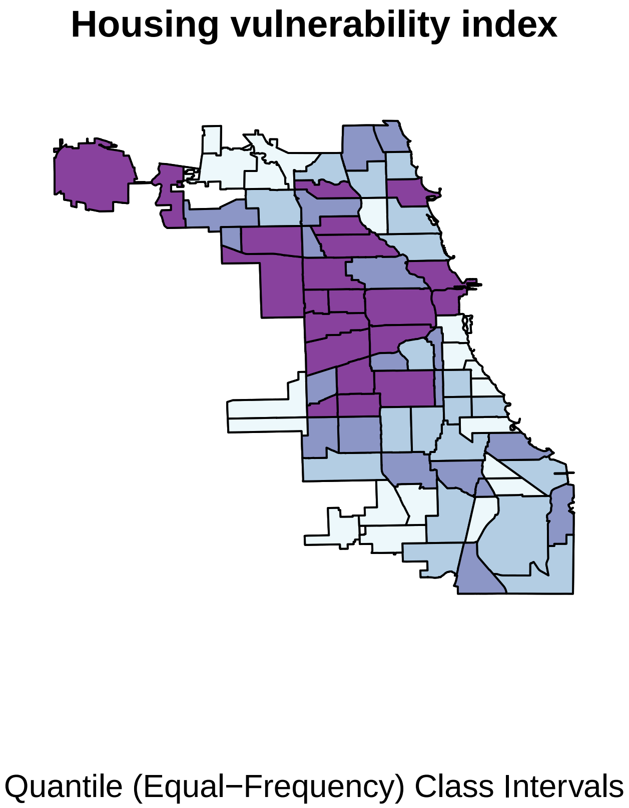
```
## K-means clustering with 4 clusters of sizes 13, 19, 20, 25
##
## Cluster means:
##   Prenatal.Care.Beginning.in.First.Trimester Preterm.Births
## 1                                 -0.3256738      0.4476259
## 2                                  0.8384152     -0.5815239
## 3                                 -0.8539439      1.1717480
## 4                                  0.2153100     -0.7282056
##   Teen.Birth.Rate Assault..Homicide. Breast.cancer.in.females
## 1       0.1904223          0.1974286                0.2464789
## 2      -1.1222707         -0.8333039               -0.0858343
## 3       1.0935036          1.4395905                0.6929625
## 4      -0.1208968         -0.6210243               -0.6173050
##   Cancer..All.Sites. Colorectal.Cancer Diabetes.related Firearm.related
## 1          0.4845801         0.5867951        0.5356863       0.1324129
## 2         -0.5140490        -0.2147842       -1.0255926      -0.7445613
## 3          1.2300010         0.9194642        1.0050860       1.4163682
## 4         -0.8453052        -0.8774687       -0.3031753      -0.6360827
##   Infant.Mortality.Rate Lung.Cancer Prostate.Cancer.in.Males
## 1             0.5048974   0.4016318                0.3594285
## 2            -0.7008346  -0.4599026               -0.5471352
## 3             1.1774213   1.1674736                1.2087680
## 4            -0.6718493  -0.7933014               -0.7380945
##   Stroke..Cerebrovascular.Disease. Childhood.Lead.Poisoning
## 1                        0.1987206               -0.2161863
## 2                       -0.6518181               -0.7062924
## 3                        1.1787218                1.0956451
## 4                       -0.5509303               -0.2273170
##   Gonorrhea.in.Females Gonorrhea.in.Males Tuberculosis   UNINSURED
## 1            0.2637136          0.3459973   -0.1689042 -0.16913450
## 2           -0.7242431         -0.7234581   -0.7919931 -1.17042172
## 3            1.3954420          1.3780558    0.1876313  0.08792185
## 4           -0.7030599         -0.7325351    0.5396399  0.90713296
##   HYPERTENSION
## 1  -0.07401297
## 2  -0.10342856
## 3   0.26947725
## 4  -0.09848935
##
## Clustering vector:
##  [1] 4 4 4 1 3 3 1 4 4 2 4 4 3 1 3 1 2 1 2 3 4 4 2 3 2 3 4 2 1 3 1 4 1 2 4
## [36] 2 2 2 2 4 4 2 4 4 4 1 2 2 2 1 1 2 3 2 2 4 1 2 3 3 4 3 3 1 4 3 4 3 3 4
## [71] 3 3 4 3 4 4 3
##
## Within cluster sum of squares by cluster:
## [1] 104.2158 129.4656 224.7091 142.1703
##  (between_SS / total_SS =  58.4 %)
##
## Available components:
##
## [1] "cluster"      "centers"      "totss"        "withinss"
## [5] "tot.withinss" "betweenss"    "size"         "iter"
## [9] "ifault"
```

```
colortab <- data.frame(cluster = hthkm1_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 = "Health indicators - k-means clusters", )
```



Housing vulnerability indicators

```
chihousing <- read.xlsx2("Housing.xlsx", sheetIndex = 1)
chihousing$PCHOMELESS <- as.numeric(as.character(chihousing$PCHOMELESS))
chihousing$PERCENT.OF.HOUSING.CROWDED <- as.numeric(as.character(chihousing$PERCENT.OF.HOUSING.CROWDED)

hovarnames <- c("PCHOMELESS", "PERCENT.OF.HOUSING.CROWDED")


hovd1 <- chihousing[, hovarnames]
hovds <- scale(hovd1)

hopcscores <- hovds[, 1] + hovds[, 2]
chihousing$pcscores <- hopcscores
housingIndex <- sapply(data.frame(chicago@data)$commareano, function(x) chihousing[chihousing$COMMAREAN
    x, ]$pcscores)
plotvar <- housingIndex
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 = "Housing vulnerability index", sub = "Quantile (Equal-Frequency) Class Intervals")
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```


Housing vulnerability index
Quantile (Equal−Frequency) Class Intervals

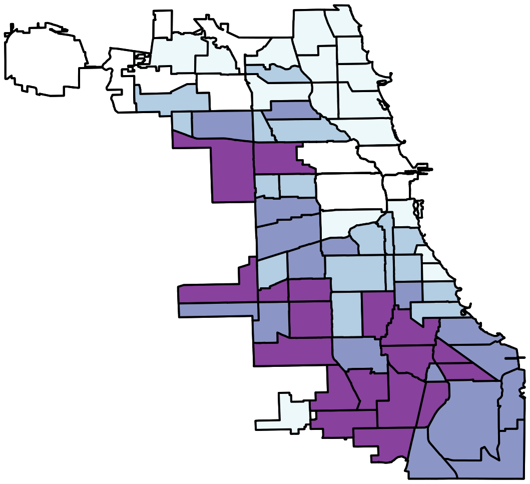
Flood frequency score

```
ziptocom <- read.xlsx2("Community area and zip code equivalency.xlsx", sheetIndex = 1)

ziptocom$COMMAREA <- as.numeric(as.character(ziptocom$COMMAREA))
ziptocom$zipCode <- as.numeric(as.character(ziptocom$zipCode))


ownAsst <- read.csv("chi_housing_assistance_owners.csv")

ownAsst <- ownAsst %>% group_by(zipCode) %>% summarise(totalApprovedIhpAmount = sum(totalApprovedIhpAmou

# ownAsst <- ownAsst %>% group_by(zipCode) %>% summarise(validRegistrations =
# sum(validRegistrations), averageFemaInspectedDamage =
# mean(averageFemaInspectedDamage), totalInspected = sum(totalInspected),
# totalDamage = sum(totalDamage), noFemaInspectedDamage =
# sum(noFemaInspectedDamage), femaInspectedDamageBetween1And10000 =
# sum(femaInspectedDamageBetween1And10000), femaInspectedDamageGreaterThan30000 =
# sum(femaInspectedDamageGreaterThan30000),
# femaInspectedDamageBetween10001And20000 =
# sum(femaInspectedDamageBetween10001And20000),
# femaInspectedDamageBetween20001And30000 =
# sum(femaInspectedDamageBetween20001And30000), approvedForFemaAssistance =
# sum(approvedForFemaAssistance), totalApprovedIhpAmount =
# sum(totalApprovedIhpAmount), repairReplaceAmount = sum(repairReplaceAmount),
# rentalAmount = sum(rentalAmount), otherNeedsAmount = sum(otherNeedsAmount),
# approvedBetween1And10000 = sum(approvedBetween1And10000),
# approvedBetween10001And25000 = sum(approvedBetween10001And25000),
# approvedBetween25001AndMax = sum(approvedBetween25001AndMax), totalMaxGrants =
# sum(totalMaxGrants))
ownAsst <- merge(x = ownAsst, y = ziptocom, by = "zipCode", all.y = TRUE)


ownAsst <- ownAsst[!is.na(ownAsst$COMMAREA), ]

ownAsst <- ownAsst %>% group_by(COMMAREA) %>% summarise(totalApprovedIhpAmount = sum(totalApprovedIhpAmo
# ownAsst <- ownAsst %>% group_by(COMMAREA) %>% summarise(validRegistrations =
# sum(validRegistrations), averageFemaInspectedDamage =
# mean(averageFemaInspectedDamage), totalInspected = sum(totalInspected),
# totalDamage = sum(totalDamage), noFemaInspectedDamage =
# sum(noFemaInspectedDamage), femaInspectedDamageBetween1And10000 =
# sum(femaInspectedDamageBetween1And10000), femaInspectedDamageGreaterThan30000 =
# sum(femaInspectedDamageGreaterThan30000),
# femaInspectedDamageBetween10001And20000 =
# sum(femaInspectedDamageBetween10001And20000),
# femaInspectedDamageBetween20001And30000 =
# sum(femaInspectedDamageBetween20001And30000), approvedForFemaAssistance =
# sum(approvedForFemaAssistance), totalApprovedIhpAmount =
# sum(totalApprovedIhpAmount), repairReplaceAmount = sum(repairReplaceAmount),
# rentalAmount = sum(rentalAmount), otherNeedsAmount = sum(otherNeedsAmount),
# approvedBetween1And10000 = sum(approvedBetween1And10000),
# approvedBetween10001And25000 = sum(approvedBetween10001And25000),
# approvedBetween25001AndMax = sum(approvedBetween25001AndMax), totalMaxGrants =
# sum(totalMaxGrants))

rentAsst <- read.csv("chi_housing_assistance_renters.csv")

rentAsst <- rentAsst %>% group_by(zipCode) %>% summarise(totalApprovedIhpAmount = sum(totalApprovedIhpAm
# rentAsst <- rentAsst %>% group_by(zipCode) %>% summarise(validRegistrations =
# sum(validRegistrations), totalInspected = sum(totalInspected),
# totalInspectedWithNoDamage = sum(totalInspectedWithNoDamage),
# totalWithModerateDamage = sum(totalWithModerateDamage), totalWithMajorDamage =
# sum(totalWithMajorDamage), totalWithSubstantialDamage =
# sum(totalWithSubstantialDamage), femaInspectedDamageBetween20001And30000 =
# sum(femaInspectedDamageBetween20001And30000), approvedForFemaAssistance =
# sum(approvedForFemaAssistance), totalApprovedIhpAmount =
# sum(totalApprovedIhpAmount), repairReplaceAmount = sum(repairReplaceAmount),
# rentalAmount = sum(rentalAmount), otherNeedsAmount = sum(otherNeedsAmount),
# approvedBetween1And10000 = sum(approvedBetween1And10000),
# approvedBetween10001And25000 = sum(approvedBetween10001And25000),
# approvedBetween25001AndMax = sum(approvedBetween25001AndMax), totalMaxGrants =
# sum(totalMaxGrants))
rentAsst <- merge(x = rentAsst, y = ziptocom, by = "zipCode", all.y = TRUE)


rentAsst <- rentAsst[!is.na(rentAsst$COMMAREA), ]
rentAsst <- rentAsst %>% group_by(COMMAREA) %>% summarise(totalApprovedIhpAmountRent = sum(totalApproved
# rentAsst <- rentAsst %>% group_by(COMMAREA) %>% summarise(validRegistrations =
# sum(validRegistrations), totalInspected = sum(totalInspected),
# totalInspectedWithNoDamage = sum(totalInspectedWithNoDamage),
# totalWithModerateDamage = sum(totalWithModerateDamage), totalWithMajorDamage =
# sum(totalWithMajorDamage), totalWithSubstantialDamage =
# sum(totalWithSubstantialDamage), approvedForFemaAssistance =
# sum(approvedForFemaAssistance), totalApprovedIhpAmountRent =
# sum(totalApprovedIhpAmount), repairReplaceAmount = sum(repairReplaceAmount),
# rentalAmount = sum(rentalAmount), otherNeedsAmount = sum(otherNeedsAmount),
# approvedBetween1And10000 = sum(approvedBetween1And10000),
# approvedBetween10001And25000 = sum(approvedBetween10001And25000),
# approvedBetween25001AndMax = sum(approvedBetween25001AndMax), totalMaxGrants =
# sum(totalMaxGrants))


asst <- merge(ownAsst, rentAsst[, c("COMMAREA", "totalApprovedIhpAmountRent")], by = "COMMAREA")
asst$TOTALAPPROVEDAMOUNT <- asst$totalApprovedIhpAmount + asst$totalApprovedIhpAmountRent


plotvar <- sapply(data.frame(chicago@data)$commareano, function(x) asst[asst$COMMAREA ==
    x, ]$TOTALAPPROVEDAMOUNT)
nclr <- 4
plotclr <- brewer.pal(nclr, "BuPu")
class <- classIntervals(plotvar, nclr, style = "quantile")
```

```
## Warning in classIntervals(plotvar, nclr, style = "quantile"): var has
## missing values, omitted in finding classes
```

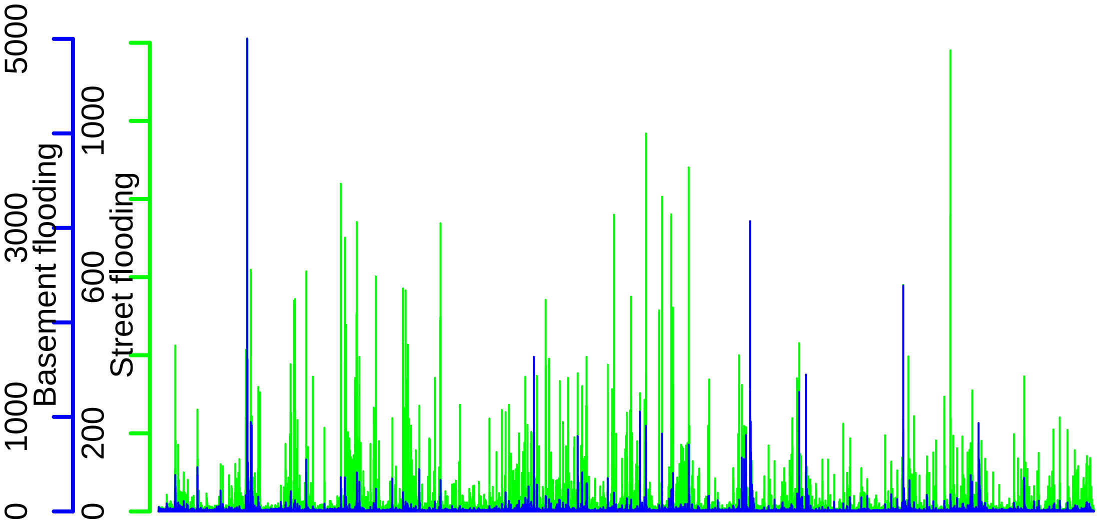
```
colcode <- findColours(class, plotclr)
plot(chicago)
plot(chicago, col = colcode, add = T)
title(main = "Flood Assistance across neighbourhoods", sub = "Quantile (Equal-Frequency) Class Intervals
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```

Flood Assistance across neighbourhoods



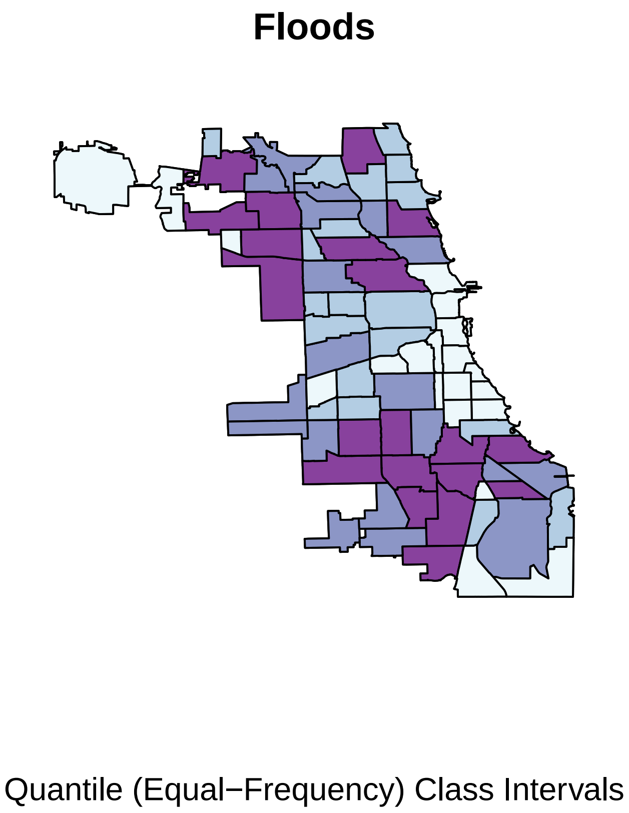
Quantile (Equal−Frequency) Class Intervals

```r
bsmt <- read.csv("wib_calls_311_comm.csv")

street <- read.csv("wos_calls_311_comm.csv")

bsmt$Created.Date <- as.Date(bsmt$Created.Date, format = "%Y-%m-%d")
bsmt$noCalls <- rowSums(bsmt[, -1])

bsmt <- arrange(bsmt, -noCalls)

street$Created.Date <- as.Date(street$Created.Date, format = "%Y-%m-%d")
street$noCalls <- rowSums(street[, -1])

street <- arrange(street, -noCalls)

plot(street$Created.Date, street$noCalls, type = "h", col = "green", axes = F, ylim = c(0,
    max(street$noCalls)), xlab = "", ylab = "")
axis(2, ylim = c(0, max(street$noCalls)), col = "green", lwd = 2, line = -0.75)
mtext(2, text = "Street flooding", line = -0.5)
par(new = T)
plot(bsmt$Created.Date, bsmt$noCalls, type = "h", col = "blue", axes = F, ylim = c(0,
    max(bsmt$noCalls)), xlab = "", ylab = "")
axis(2, ylim = c(0, max(bsmt$noCalls)), lwd = 2, line = 1.25, col = "blue")
mtext(2, text = "Basement flooding", line = 1.5)
```

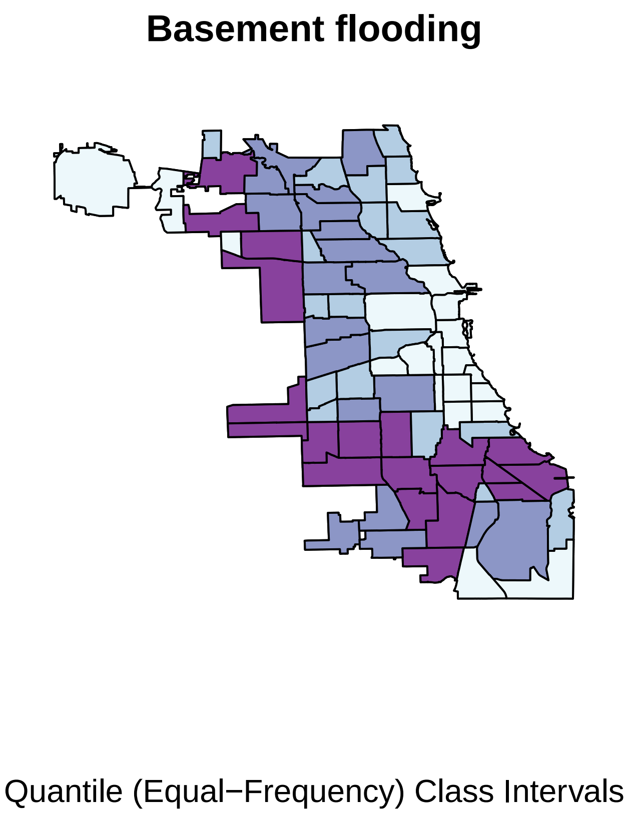


```
bsmtPks <- bsmt[1:201, ]
streetPks <- street[1:321, ]

floods <- data.frame(chicago@data[, tolower(c("COMMAREA", "COMMAREANO"))])



floods$NOEVENTS <- 0

fillFloodCount <- function(comArea, val) {
    floods[floods$commarea == comArea, ]$NOEVENTS <- val
}

getFloodCount <- function(comArea) {
    comArea <- gsub(" ", ".", gsub("'", "", trimws(comArea)))
    return(sum(with(bsmtPks, bsmtPks[, c(comArea)])) + sum(with(streetPks, streetPks[,
        c(comArea)])))
}
getBsmtCount <- function(comArea) {
    comArea <- gsub(" ", ".", gsub("'", "", trimws(comArea)))
    return(sum(with(bsmtPks, bsmtPks[, c(comArea)])))
}
getStreetFloodCount <- function(comArea) {
    comArea <- gsub(" ", ".", gsub("'", "", trimws(comArea)))
    return(sum(with(streetPks, streetPks[, c(comArea)])))
}
bsmtEvents <- floods
streetEvents <- floods
floods$NOEVENTS <- sapply(floods$commarea, getFloodCount)
bsmtEvents$NOEVENTS <- sapply(bsmtEvents$commarea, getBsmtCount)
streetEvents$NOEVENTS <- sapply(streetEvents$commarea, getStreetFloodCount)

floods$score <- scale(floods$NOEVENTS)

floodscore <- sapply(data.frame(chicago@data)$commareano, function(x) floods[floods$commareano ==
    x, ]$score)
```

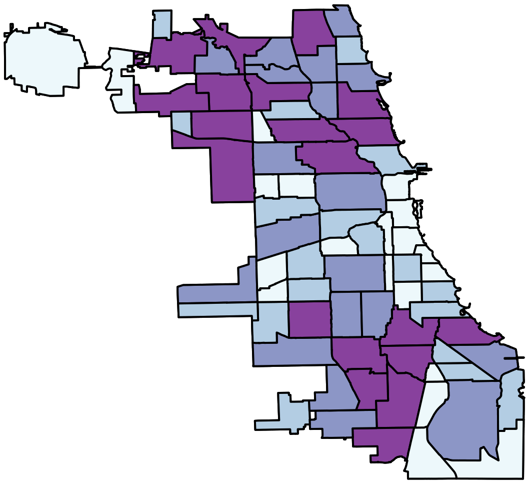
```
plotvar <- floodscore
nclr <- 4
plotclr <- brewer.pal(nclr, "BuPu")
class <- classIntervals(plotvar, nclr, style = "quantile")
colcode <- findColours(class, plotclr)
plot(chicago)
plot(chicago, col = colcode, add = T)
title(main = "Floods", sub = "Quantile (Equal-Frequency) Class Intervals")
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```



```
bsmtEvents$score <- scale(bsmtEvents$NOEVENTS)
bsmtscore <- sapply(data.frame(chicago@data)$commareano, function(x) bsmtEvents[bsmtEvents$commareano ==
    x, ]$score)
plotvar <- bsmtscore
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 = "Basement flooding", sub = "Quantile (Equal-Frequency) Class Intervals")
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```



```
streetEvents$score <- scale(streetEvents$NOEVENTS)
streetscore <- sapply(data.frame(chicago@data)$commareano, function(x) streetEvents[streetEvents$commare
    x, ]$score)
plotvar <- streetscore
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 = "Street flooding", sub = "Quantile (Equal-Frequency) Class Intervals")
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```

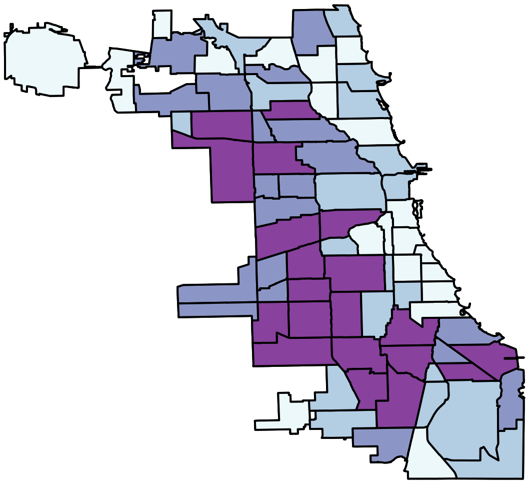


Street flooding

Quantile (Equal−Frequency) Class Intervals

```r
finalScore <- 0.2 * scale(ecoIndex) + 0.1 * scale(healthIndex) + 0.2 * scale(housingIndex) +
    0.5 * scale(bsmtscore)

plotvar <- finalScore
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 = "Street flooding", sub = "Quantile (Equal-Frequency) Class Intervals")
legend(100, 44, legend = names(attr(colcode, "table")), fill = attr(colcode, "palette"),
    cex = 0.6, bty = "n")
```

Street flooding



Quantile (Equal−Frequency) Class Intervals