Covid-Alcohol-Consumption

Efeler

01/12/2021

Orçun S. Tandoğan 2535037

Emre Özzeybek 2099661

Source: https://bit.io/bitdotio/iowa_liquor_unemployment (https://bit.io/bitdotio/iowa_liquor_unemployment)

```
#drops <- c( "cases",'deaths')
#dataset <- dataset[ , !(names(dataset) %in% drops)]
dataset$cases[is.na(dataset$cases)] <- 0
dataset$deaths[is.na(dataset$deaths)] <- 0
dataset <- na.omit(dataset)
dataset <- merge(dataset, secondary_dataset[,c("year_month","time_index")], by = "year_month")
#rm(drops)
```

Introducing the data

This dataset has 2 tables: The first one shows the state of employment and liquor consumption in the state of Iowa from 2012 to April 2021. The second one of various house listings in London and neighboring regions. It has a sample size of 3481 and has the 11 features listed below:

Age: Median age **Race, White:** Percent of population identifying as white **Race, Black:** Percent of population identifying as black **Race, Asian:** Percent of population identifying as Asian **Income:** Median Income **Health Insured:** Percent of population with health insurance **Poverty:** Percent of population living below the poverty line **Graduate Degrees:** Percent of population with professional or graduate degrees **COVID Cases:** Number of COVID cases during first 12 months of pandemic divided by population **COVID Deaths:** Number of COVID deaths during first 12 months of pandemic divided by population **Population Density:** Log of population per square mile

Determining Numerical & Categorical Variables

First we determine the numerical and the categorical data so we can deal with them accordingly.

```
dn <- dataset[ , unlist(lapply(dataset, is.numeric))]
details(dn)
```

```
## [1] "-------------------------------------------------------------------------------------------------------"
## [1] "Column Names Of Numeric Based Variables: "
##  [1] "fips"                  "n_liters"              "n_transactions"
##  [4] "n_bottles"             "bottles_p_transaction" "employment"
##  [7] "unemployment"          "unemployment_rate"     "cases"
## [10] "deaths"                "pop_total"             "median_age"
## [13] "pop_gt18"              "pop_gt21"              "perc_gt21"
## [16] "perc_race_white"       "perc_race_black"       "perc_race_asian"
## [19] "perc_race_multi"       "median_income"         "perc_snap"
## [22] "perc_health_insurance" "perc_poverty"          "perc_ed_gt_hs"
## [25] "perc_ed_gt_bach"       "perc_ed_gt_grad"       "county_num"
## [28] "area_sq_mi"            "lat"                   "long"
## [31] "time_index"
## [1] "-------------------------------------------------------------------------------------------------------"
## [1] "Number Of Numeric Based Variables: 31"
```

```
# Creating Unique & NA Values Dataframe
dn_una <- data.frame(matrix(ncol = 0 , nrow = 5))
add_column_index(f_und(dn,dn_una,0))
```

```
##                                                      fips  n_liters
## Column_Index                                        1.000     2.000
## Number Of Unique Values / Total Number Of Values:   0.009     0.998
## Number Of Unique Values:                           99.000 10949.000
## Number Of NA Values / Total Number Of Values:       0.000     0.000
## Number Of NA Values:                                0.000     0.000
## Total Number Of Values:                         10972.000 10972.000
##                                                 n_transactions n_bottles
## Column_Index                                             3.000     4.000
## Number Of Unique Values / Total Number Of Values:        0.318     0.762
## Number Of Unique Values:                              3489.000  8362.000
## Number Of NA Values / Total Number Of Values:            0.000     0.000
## Number Of NA Values:                                     0.000     0.000
## Total Number Of Values:                              10972.000 10972.000
##                                                 bottles_p_transaction
## Column_Index                                                    5.000
## Number Of Unique Values / Total Number Of Values:               0.987
## Number Of Unique Values:                                    10833.000
## Number Of NA Values / Total Number Of Values:                   0.000
## Number Of NA Values:                                            0.000
## Total Number Of Values:                                     10972.000
##                                                 employment unemployment
## Column_Index                                         6.000        7.000
## Number Of Unique Values / Total Number Of Values:    0.203        0.027
## Number Of Unique Values:                          2225.000      292.000
## Number Of NA Values / Total Number Of Values:        0.000        0.000
## Number Of NA Values:                                 0.000        0.000
## Total Number Of Values:                          10972.000    10972.000
##                                                 unemployment_rate     cases
## Column_Index                                                8.000     9.000
## Number Of Unique Values / Total Number Of Values:           0.011     0.045
## Number Of Unique Values:                                  122.000   489.000
## Number Of NA Values / Total Number Of Values:               0.000     0.000
## Total Number Of Values:                                     0.000     0.000
## Total Number Of Values:                                 10972.000 10972.000
##                                                    deaths pop_total
## Column_Index                                       10.000    11.000
## Number Of Unique Values / Total Number Of Values:   0.005     0.009
## Number Of Unique Values:                           54.000    99.000
## Number Of NA Values / Total Number Of Values:       0.000     0.000
## Number Of NA Values:                                0.000     0.000
## Total Number Of Values:                         10972.000 10972.000
##                                                 median_age  pop_gt18
## Column_Index                                        12.000    13.000
## Number Of Unique Values / Total Number Of Values:    0.006     0.009
## Number Of Unique Values:                            69.000    99.000
## Number Of NA Values / Total Number Of Values:        0.000     0.000
## Number Of NA Values:                                 0.000     0.000
## Total Number Of Values:                          10972.000 10972.000
##                                                  pop_gt21 perc_gt21
## Column_Index                                       14.000    15.000
## Number Of Unique Values / Total Number Of Values:   0.009     0.005
## Number Of Unique Values:                           99.000    56.000
## Number Of NA Values / Total Number Of Values:       0.000     0.000
## Number Of NA Values:                                0.000     0.000
## Total Number Of Values:                         10972.000 10972.000
##                                                                                                  Row_Names_Middle
## Column_Index                                                                                                   16
## Number Of Unique Values / Total Number Of Values: Number Of Unique Values / Total Number Of Values:
## Number Of Unique Values:                                                          Number Of Unique Values:
## Number Of NA Values / Total Number Of Values:          Number Of NA Values / Total Number Of Values:
## Number Of NA Values:                                                                 Number Of NA Values:
## Total Number Of Values:                                                           Total Number Of Values:
##                                                 perc_race_white
## Column_Index                                          17.000
## Number Of Unique Values / Total Number Of Values:      0.005
## Number Of Unique Values:                              56.000
## Number Of NA Values / Total Number Of Values:          0.000
## Number Of NA Values:                                   0.000
## Total Number Of Values:                            10972.000
##                                                 perc_race_black
## Column_Index                                          18.000
## Number Of Unique Values / Total Number Of Values:      0.003
```

```
## Number Of Unique Values:                                  36.000
## Number Of NA Values / Total Number Of Values:              0.000
## Number Of NA Values:                                       0.000
## Total Number Of Values:                                10972.000
##                                                  perc_race_asian
## Column_Index                                              19.000
## Number Of Unique Values / Total Number Of Values:          0.003
## Number Of Unique Values:                                  31.000
## Number Of NA Values / Total Number Of Values:              0.000
## Number Of NA Values:                                       0.000
## Total Number Of Values:                                10972.000
##                                                  perc_race_multi median_income
## Column_Index                                              20.000        21.000
## Number Of Unique Values / Total Number Of Values:          0.003         0.009
## Number Of Unique Values:                                  29.000        99.000
## Number Of NA Values / Total Number Of Values:              0.000         0.000
## Number Of NA Values:                                       0.000         0.000
## Total Number Of Values:                                10972.000     10972.000
##                                                  perc_snap
## Column_Index                                        22.000
## Number Of Unique Values / Total Number Of Values:    0.006
## Number Of Unique Values:                            66.000
## Number Of NA Values / Total Number Of Values:        0.000
## Number Of NA Values:                                 0.000
## Total Number Of Values:                          10972.000
##                                                  perc_health_insurance
## Column_Index                                                    23.000
## Number Of Unique Values / Total Number Of Values:                0.004
## Number Of Unique Values:                                        47.000
## Number Of NA Values / Total Number Of Values:                    0.000
## Number Of NA Values:                                             0.000
## Total Number Of Values:                                      10972.000
##                                                  perc_poverty perc_ed_gt_hs
## Column_Index                                        24.000        25.000
## Number Of Unique Values / Total Number Of Values:    0.006         0.006
## Number Of Unique Values:                            61.000        62.000
## Number Of NA Values / Total Number Of Values:        0.000         0.000
## Number Of NA Values:                                 0.000         0.000
## Total Number Of Values:                          10972.000     10972.000
##                                                  perc_ed_gt_bach
## Column_Index                                              26.000
## Number Of Unique Values / Total Number Of Values:          0.007
## Number Of Unique Values:                                  76.000
## Number Of NA Values / Total Number Of Values:              0.000
## Number Of NA Values:                                       0.000
## Total Number Of Values:                                10972.000
##                                                  perc_ed_gt_grad county_num
## Column_Index                                              27.000     28.000
## Number Of Unique Values / Total Number Of Values:          0.005      0.009
## Number Of Unique Values:                                  50.000     99.000
## Number Of NA Values / Total Number Of Values:              0.000      0.000
## Number Of NA Values:                                       0.000      0.000
## Total Number Of Values:                                10972.000  10972.000
##                                                  area_sq_mi       lat
## Column_Index                                         29.000    30.000
## Number Of Unique Values / Total Number Of Values:     0.009     0.009
## Number Of Unique Values:                             99.000    99.000
## Number Of NA Values / Total Number Of Values:         0.000     0.000
## Number Of NA Values:                                  0.000     0.000
## Total Number Of Values:                           10972.000 10972.000
##                                                      long
## Column_Index                                        31.000
## Number Of Unique Values / Total Number Of Values:    0.009
## Number Of Unique Values:                            99.000
## Number Of NA Values / Total Number Of Values:        0.000
## Number Of NA Values:                                 0.000
## Total Number Of Values:                          10972.000
##                                                                                                  Row_Names_End
## Column_Index                                                                                                32
## Number Of Unique Values / Total Number Of Values: Number Of Unique Values / Total Number Of Values:
## Number Of Unique Values:                                                                 Number Of Unique Values:
## Number Of NA Values / Total Number Of Values:         Number Of NA Values / Total Number Of Values:
## Number Of NA Values:                                                                        Number Of NA Values:
## Total Number Of Values:                                                                   Total Number Of Values:
```

```
# Head
head( add_column_index(dn[complete.cases(dn), ]) ) # dataset[complete.cases(dataset), ] It allows to get rid of NAs
```

```
##               fips n_liters n_transactions n_bottles bottles_p_transaction
## Column_Index     1     2.00              3         4              5.000000
## 1            19001  2165.94            319      2263              7.094044
## 2            19003   701.04            169       686              4.059172
## 3            19005  4177.72            678      3933              5.800885
## 4            19007  3395.90            620      3571              5.759677
## 5            19009  1245.01            199      1188              5.969849
##              employment unemployment unemployment_rate cases deaths pop_total
## Column_Index          6            7               8.0     9     10        11
## 1                  3820          210               5.1     0      0      7085
## 2                  2110          110               4.9     0      0      3670
## 3                  7010          690               8.9     0      0     13813
## 4                  5520          460               7.6     0      0     12452
## 5                  3070          210               6.5     0      0      5571
##              median_age pop_gt18 pop_gt21 perc_gt21 perc_race_white
## Column_Index       12.0       13       14      15.0            16.0
## 1                  45.4     5566     5397      76.2            97.5
## 2                  47.3     2903     2813      76.6            98.6
## 3                  44.4    10619    10254      74.2            96.5
## 4                  45.7     9719     9308      74.8            97.0
## 5                  48.1     4425     4262      76.5            96.9
##              perc_race_black perc_race_asian perc_race_multi median_income
## Column_Index            17.0            18.0            19.0            20
## 1                        0.6             0.7             1.0         53363
## 2                        0.1             0.2             0.2         49255
## 3                        1.3             0.6             1.0         52216
## 4                        0.7             0.5             1.5         40167
## 5                        0.2             0.3             2.5         52055
##              perc_snap perc_health_insurance perc_poverty perc_ed_gt_hs
## Column_Index      21.0                  22.0         23.0          24.0
## 1                 11.7                  95.6          5.8          94.2
## 2                 10.4                  94.2          7.1          93.8
## 3                  9.4                  92.4          7.4          88.3
## 4                 14.4                  92.2         11.4          90.2
## 5                  8.3                  94.8          8.6          89.5
##              perc_ed_gt_bach perc_ed_gt_grad county_num area_sq_mi      lat
## Column_Index            25.0            26.0         27     28.000 29.00000
## 1                       18.5             4.2          1    569.271 41.32853
## 2                       15.6             3.6          2    423.433 41.02166
## 3                       17.8             4.7          3    639.044 43.27496
## 4                       16.3             4.8          4    497.289 40.74425
## 5                       17.1             5.1          5    442.961 41.67918
##                  long time_index
## Column_Index  30.00000         31
## 1            -94.47816          0
## 2            -94.69691          0
## 3            -91.38275          0
## 4            -92.87306          0
## 5            -94.90431          0
```

```
dc <- dataset[ , unlist(lapply(dataset, is.character))]
details(dc)
```

```
## [1] "-------------------------------------------------------------------------------------------------------"
## [1] "Column Names Of Numeric Based Variables: "
## [1] "year_month" "county"
## [1] "-------------------------------------------------------------------------------------------------------"
## [1] "Number Of Numeric Based Variables: 2"
```

```
# Creating Unique & NA Values Dataframe
dc_una <- data.frame(matrix(ncol = 0 , nrow = 5))
add_column_index(f_und(dc,dc_una,0))
```

```
##                                                year_month
## Column_Index                                         1.00
## Number Of Unique Values / Total Number Of Values:    0.01
## Number Of Unique Values:                           111.00
## Number Of NA Values / Total Number Of Values:        0.00
## Number Of NA Values:                                 0.00
## Total Number Of Values:                          10972.00
##                                                                                  Row_Names_Middle
## Column_Index                                                                                    2
## Number Of Unique Values / Total Number Of Values: Number Of Unique Values / Total Number Of Values:
## Number Of Unique Values:                                                      Number Of Unique Values:
## Number Of NA Values / Total Number Of Values:         Number Of NA Values / Total Number Of Values:
## Number Of NA Values:                                                             Number Of NA Values:
## Total Number Of Values:                                                       Total Number Of Values:
##                                                  county
## Column_Index                                      3.000
## Number Of Unique Values / Total Number Of Values: 0.009
## Number Of Unique Values:                         99.000
## Number Of NA Values / Total Number Of Values:     0.000
## Number Of NA Values:                              0.000
## Total Number Of Values:                       10972.000
##                                                                                    Row_Names_End
## Column_Index                                                                                   4
## Number Of Unique Values / Total Number Of Values: Number Of Unique Values / Total Number Of Values:
## Number Of Unique Values:                                                      Number Of Unique Values:
## Number Of NA Values / Total Number Of Values:         Number Of NA Values / Total Number Of Values:
## Number Of NA Values:                                                             Number Of NA Values:
## Total Number Of Values:                                                       Total Number Of Values:
```

```
# Head
head( add_column_index(dc[complete.cases(dc), ]) )
```

```
##              year_month    county
## Column_Index          1         2
## 1               2012-01     adair
## 2               2012-01     adams
## 3               2012-01 allamakee
## 4               2012-01 appanoose
## 5               2012-01   audubon
```

```
rm(dc,dc_una)
```

Analyze Unique Values

```
numerical_columns <- colnames(dataset[ , unlist(lapply(dataset, is.numeric))]  )

categorical_columns <- colnames(dataset[ , unlist(lapply(dataset, is.character))])
# Creating A Dataset
# Still consists categorical ones with characters
Numeric_Dataset <- dataset[c(numerical_columns,categorical_columns)]
dn_na <- data.frame(matrix(ncol = 0 , nrow = 3))

add_column_index(f_und(Numeric_Dataset,dn_na,1))
```

```
##                                            fips n_liters n_transactions
## Column_Index                                  1        2              3
## Number Of NA Values / Total Number Of Values:     0        0              0
## Number Of NA Values:                          0        0              0
## Total Number Of Values:                   10972    10972          10972
##                                           n_bottles bottles_p_transaction
## Column_Index                                      4                      5
## Number Of NA Values / Total Number Of Values:     0                      0
## Number Of NA Values:                              0                      0
## Total Number Of Values:                       10972                  10972
##                                           employment unemployment
## Column_Index                                       6            7
## Number Of NA Values / Total Number Of Values:      0            0
## Number Of NA Values:                               0            0
## Total Number Of Values:                        10972        10972
##                                           unemployment_rate cases deaths
## Column_Index                                              8     9     10
## Number Of NA Values / Total Number Of Values:             0     0      0
## Number Of NA Values:                                      0     0      0
## Total Number Of Values:                               10972 10972  10972
```

```
##                                                 pop_total median_age pop_gt18
## Column_Index                                           11         12       13
## Number Of NA Values / Total Number Of Values:           0          0        0
## Number Of NA Values:                                    0          0        0
## Total Number Of Values:                             10972      10972    10972
##                                                 pop_gt21 perc_gt21
## Column_Index                                           14        15
## Number Of NA Values / Total Number Of Values:           0         0
## Number Of NA Values:                                    0         0
## Total Number Of Values:                             10972     10972
##                                                 perc_race_white
## Column_Index                                                 16
## Number Of NA Values / Total Number Of Values:                 0
## Number Of NA Values:                                          0
## Total Number Of Values:                                   10972
##                                                                                                 Row_Names_Middle
## Column_Index                                                                                                  17
## Number Of NA Values / Total Number Of Values: Number Of NA Values / Total Number Of Values:
## Number Of NA Values:                                                                    Number Of NA Values:
## Total Number Of Values:                                                               Total Number Of Values:
##                                                 perc_race_black perc_race_asian
## Column_Index                                                 18               19
## Number Of NA Values / Total Number Of Values:                 0                0
## Number Of NA Values:                                          0                0
## Total Number Of Values:                                   10972            10972
##                                                 perc_race_multi median_income
## Column_Index                                                 20            21
## Number Of NA Values / Total Number Of Values:                 0             0
## Number Of NA Values:                                          0             0
## Total Number Of Values:                                   10972         10972
##                                                 perc_snap perc_health_insurance
## Column_Index                                           22                    23
## Number Of NA Values / Total Number Of Values:           0                     0
## Number Of NA Values:                                    0                     0
## Total Number Of Values:                             10972                 10972
##                                                 perc_poverty perc_ed_gt_hs
## Column_Index                                              24            25
## Number Of NA Values / Total Number Of Values:              0             0
## Number Of NA Values:                                       0             0
## Total Number Of Values:                                10972         10972
##                                                 perc_ed_gt_bach perc_ed_gt_grad
## Column_Index                                                 26               27
## Number Of NA Values / Total Number Of Values:                 0                0
## Number Of NA Values:                                          0                0
## Total Number Of Values:                                   10972            10972
##                                                 county_num area_sq_mi   lat  long
## Column_Index                                            28         29    30    31
## Number Of NA Values / Total Number Of Values:            0          0     0     0
## Number Of NA Values:                                     0          0     0     0
## Total Number Of Values:                              10972      10972 10972 10972
##                                                 time_index year_month
## Column_Index                                            32         33
## Number Of NA Values / Total Number Of Values:            0          0
## Number Of NA Values:                                     0          0
## Total Number Of Values:                              10972      10972
##                                                                                                   Row_Names_End
## Column_Index                                                                                                 34
## Number Of NA Values / Total Number Of Values: Number Of NA Values / Total Number Of Values:
## Number Of NA Values:                                                                      Number Of NA Values:
## Total Number Of Values:                                                                 Total Number Of Values:
```

```
head(Numeric_Dataset)
```

```
##    fips n_liters n_transactions n_bottles bottles_p_transaction employment
## 1 19001  2165.94            319      2263              7.094044       3820
## 2 19003   701.04            169       686              4.059172       2110
## 3 19005  4177.72            678      3933              5.800885       7010
## 4 19007  3395.90            620      3571              5.759677       5520
## 5 19009  1245.01            199      1188              5.969849       3070
## 6 19011  3474.91            567      3403              6.001764      13100
##   unemployment unemployment_rate cases deaths pop_total median_age pop_gt18
## 1          210               5.1     0      0      7085       45.4     5566
## 2          110               4.9     0      0      3670       47.3     2903
## 3          690               8.9     0      0     13813       44.4    10619
## 4          460               7.6     0      0     12452       45.7     9719
## 5          210               6.5     0      0      5571       48.1     4425
## 6          900               6.6     0      0     25636       42.6    19584
##   pop_gt21 perc_gt21 perc_race_white perc_race_black perc_race_asian
## 1     5397      76.2            97.5             0.6             0.7
## 2     2813      76.6            98.6             0.1             0.2
## 3    10254      74.2            96.5             1.3             0.6
## 4     9308      74.8            97.0             0.7             0.5
## 5     4262      76.5            96.9             0.2             0.3
## 6    18843      73.5            96.9             0.6             0.3
##   perc_race_multi median_income perc_snap perc_health_insurance perc_poverty
## 1             1.0         53363      11.7                  95.6          5.8
## 2             0.2         49255      10.4                  94.2          7.1
## 3             1.0         52216       9.4                  92.4          7.4
## 4             1.5         40167      14.4                  92.2         11.4
## 5             2.5         52055       8.3                  94.8          8.6
## 6             1.3         67729       8.2                  97.8          6.2
##   perc_ed_gt_hs perc_ed_gt_bach perc_ed_gt_grad county_num area_sq_mi      lat
## 1          94.2            18.5             4.2          1    569.271 41.32853
## 2          93.8            15.6             3.6          2    423.433 41.02166
## 3          88.3            17.8             4.7          3    639.044 43.27496
## 4          90.2            16.3             4.8          4    497.289 40.74425
## 5          89.5            17.1             5.1          5    442.961 41.67918
## 6          94.1            22.1             6.4          6    716.102 42.09255
##        long time_index year_month    county
## 1 -94.47816          0    2012-01     adair
## 2 -94.69691          0    2012-01     adams
## 3 -91.38275          0    2012-01 allamakee
## 4 -92.87306          0    2012-01 appanoose
## 5 -94.90431          0    2012-01   audubon
## 6 -92.05763          0    2012-01    benton
```

```
#  This is the summary of the data before preprocessing.
print( summary(dataset) )
```

```
##   year_month             fips          n_liters        n_transactions
##  Length:10972       Min.   :19001   Min.   :    56.9   Min.   :    5
##  Class :character   1st Qu.:19049   1st Qu.:  2983.9   1st Qu.:  411
##  Mode  :character   Median :19099   Median :  5887.1   Median :  768
##                     Mean   :19099   Mean   : 17336.1   Mean   : 1897
##                     3rd Qu.:19149   3rd Qu.: 12061.0   3rd Qu.: 1473
##                     Max.   :19197   Max.   :478106.7   Max.   :48060
##    n_bottles      bottles_p_transaction   employment      unemployment
##  Min.   :    76   Min.   : 2.443        Min.   :  1870   Min.   :   30.0
##  1st Qu.:  3009   1st Qu.: 6.645        1st Qu.:  5140   1st Qu.:  180.0
##  Median :  6068   Median : 8.121        Median :  7900   Median :  300.0
##  Mean   : 19724   Mean   : 8.401        Mean   : 16430   Mean   :  668.5
##  3rd Qu.: 12256   3rd Qu.: 9.908        3rd Qu.: 13400   3rd Qu.:  550.0
##  Max.   :656998   Max.   :32.143        Max.   :269800   Max.   :35600.0
##  unemployment_rate     cases              deaths          pop_total
##  Min.   : 0.90     Min.   :    0.00   Min.   :-4.0000   Min.   :  3670
##  1st Qu.: 2.80     1st Qu.:    0.00   1st Qu.: 0.0000   1st Qu.:  9802
##  Median : 3.60     Median :    0.00   Median : 0.0000   Median : 15268
##  Mean   : 3.84     Mean   :   31.93   Mean   : 0.5234   Mean   : 31752
##  3rd Qu.: 4.60     3rd Qu.:    0.00   3rd Qu.: 0.0000   3rd Qu.: 25068
##  Max.   :17.80     Max.   :13482.00   Max.   :91.0000   Max.   :479612
##    median_age       pop_gt18         pop_gt21        perc_gt21
##  Min.   :26.60   Min.   :  2903   Min.   :  2813   Min.   :66.70
##  1st Qu.:39.70   1st Qu.:  7681   1st Qu.:  7414   1st Qu.:72.20
##  Median :42.70   Median : 12049   Median : 11533   Median :73.80
##  Mean   :41.89   Mean   : 24382   Mean   : 22972   Mean   :73.45
##  3rd Qu.:44.20   3rd Qu.: 19326   3rd Qu.: 17964   3rd Qu.:74.90
##  Max.   :48.40   Max.   :359784   Max.   :341943   Max.   :80.40
##  perc_race_white perc_race_black perc_race_asian  perc_race_multi
##  Min.   :77.00   Min.   :0.000   Min.   : 0.000   Min.   :0.100
##  1st Qu.:93.70   1st Qu.:0.400   1st Qu.: 0.300   1st Qu.:1.000
##  Median :95.70   Median :1.000   Median : 0.600   Median :1.400
##  Mean   :94.25   Mean   :1.557   Mean   : 1.146   Mean   :1.485
##  3rd Qu.:97.00   3rd Qu.:2.000   3rd Qu.: 1.200   3rd Qu.:1.900
##  Max.   :99.10   Max.   :9.500   Max.   :10.100   Max.   :3.800
##  median_income     perc_snap      perc_health_insurance  perc_poverty
##  Min.   :40167   Min.   : 4.600   Min.   :73.60         Min.   : 2.500
##  1st Qu.:52323   1st Qu.: 8.100   1st Qu.:94.30         1st Qu.: 5.800
##  Median :56437   Median : 9.700   Median :95.60         Median : 7.100
##  Mean   :57263   Mean   : 9.947   Mean   :95.04         Mean   : 7.172
##  3rd Qu.:61183   3rd Qu.:11.800   3rd Qu.:96.40         3rd Qu.: 8.500
##  Max.   :88479   Max.   :17.400   Max.   :98.10         Max.   :13.300
##  perc_ed_gt_hs   perc_ed_gt_bach perc_ed_gt_grad     county
##  Min.   :78.40   Min.   :12.70   Min.   : 2.600   Length:10972
##  1st Qu.:90.40   1st Qu.:17.90   1st Qu.: 4.600   Class :character
##  Median :92.40   Median :19.80   Median : 5.500   Mode  :character
##  Mean   :91.64   Mean   :21.96   Mean   : 6.488
##  3rd Qu.:93.30   3rd Qu.:23.60   3rd Qu.: 6.900
##  Max.   :97.20   Max.   :52.80   Max.   :25.100
##    county_num      area_sq_mi         lat             long
##  Min.   : 1.00   Min.   :380.5   Min.   :40.65   Min.   :-96.22
##  1st Qu.:25.00   1st Qu.:473.2   1st Qu.:41.33   1st Qu.:-94.70
##  Median :50.00   Median :569.6   Median :42.04   Median :-93.47
##  Mean   :50.03   Mean   :564.3   Mean   :42.03   Mean   :-93.46
##  3rd Qu.:75.00   3rd Qu.:587.6   3rd Qu.:42.74   3rd Qu.:-92.17
##  Max.   :99.00   Max.   :972.7   Max.   :43.39   Max.   :-90.53
##    time_index
##  Min.   :  0.00
##  1st Qu.: 27.00
##  Median : 55.00
##  Mean   : 55.01
##  3rd Qu.: 83.00
##  Max.   :110.00
```

```
rm(dn,dn_una,dn_na,categorical_columns,numerical_columns)
```

Preprocessing

```
# We omit missing values
dataset <- na.omit(dataset)
```

Outlier handling

We get rid of some of the outliers to smooth out the clusters.

```
outlier_switch = 1

if(outlier_switch == 1){
for( i in seq.int(1,5)){
  full_list <- as.numeric(unlist(dataset[i]))
  outliers <- boxplot(full_list, plot=FALSE)$out
  temp_dataset <- dataset
  if(length(outliers) != 0){
    dataset <- temp_dataset[-which(full_list %in% outliers),]
  }
}
  rm(temp_dataset)
} else {
  dataset <- dataset
}
print( summary(dataset) )
```

```
##   year_month             fips          n_liters        n_transactions
##  Length:8893        Min.   :19001   Min.   :   56.85   Min.   :   5.0
##  Class :character   1st Qu.:19047   1st Qu.: 2530.81   1st Qu.: 349.0
##  Mode  :character   Median :19095   Median : 4563.79   Median : 636.0
##                     Mean   :19096   Mean   : 5506.06   Mean   : 711.7
##                     3rd Qu.:19143   3rd Qu.: 7855.68   3rd Qu.: 972.0
##                     Max.   :19197   Max.   :19852.40   Max.   :2232.0
##    n_bottles     bottles_p_transaction   employment     unemployment
##  Min.   :   76   Min.   : 2.443        Min.   : 1870   Min.   :  30
##  1st Qu.: 2553   1st Qu.: 6.347        1st Qu.: 4760   1st Qu.: 160
##  Median : 4698   Median : 7.575        Median : 6580   Median : 250
##  Mean   : 5602   Mean   : 7.821        Mean   : 7673   Mean   : 302
##  3rd Qu.: 7967   3rd Qu.: 9.088        3rd Qu.: 9880   3rd Qu.: 390
##  Max.   :17249   Max.   :32.143        Max.   :42000   Max.   :1700
##  unemployment_rate     cases             deaths          pop_total
##  Min.   : 0.900    Min.   :   0.00   Min.   :-4.0000   Min.   : 3670
##  1st Qu.: 2.800    1st Qu.:   0.00   1st Qu.: 0.0000   1st Qu.: 9360
##  Median : 3.600    Median :   0.00   Median : 0.0000   Median :12739
##  Mean   : 3.768    Mean   :  13.07   Mean   : 0.2836   Mean   :14732
##  3rd Qu.: 4.500    3rd Qu.:   0.00   3rd Qu.: 0.0000   3rd Qu.:18148
##  Max.   :17.800    Max.   :1771.00   Max.   :38.0000   Max.   :87099
##    median_age       pop_gt18        pop_gt21       perc_gt21
##  Min.   :33.40   Min.   : 2903   Min.   : 2813   Min.   :66.70
##  1st Qu.:41.50   1st Qu.: 7059   1st Qu.: 6751   1st Qu.:72.50
##  Median :43.00   Median : 9719   Median : 9403   Median :74.00
##  Mean   :42.75   Mean   :11340   Mean   :10804   Mean   :73.69
##  3rd Qu.:44.40   3rd Qu.:14202   3rd Qu.:13476   3rd Qu.:75.00
##  Max.   :48.40   Max.   :62763   Max.   :60554   Max.   :80.40
##  perc_race_white perc_race_black perc_race_asian   perc_race_multi
##  Min.   :77.00   Min.   :0.00    Min.   : 0.0000   Min.   :0.100
##  1st Qu.:94.90   1st Qu.:0.30    1st Qu.: 0.3000   1st Qu.:0.900
##  Median :96.40   Median :0.70    Median : 0.5000   Median :1.200
##  Mean   :95.38   Mean   :1.01    Mean   : 0.8484   Mean   :1.294
##  3rd Qu.:97.30   3rd Qu.:1.40    3rd Qu.: 1.0000   3rd Qu.:1.700
##  Max.   :99.10   Max.   :7.20    Max.   :10.1000   Max.   :3.800
##  median_income     perc_snap      perc_health_insurance  perc_poverty
##  Min.   :40167   Min.   : 4.600   Min.   :73.60         Min.   : 2.500
##  1st Qu.:52323   1st Qu.: 8.000   1st Qu.:94.10         1st Qu.: 5.600
##  Median :56037   Median : 9.600   Median :95.60         Median : 6.800
##  Mean   :56792   Mean   : 9.696   Mean   :94.95         Mean   : 7.017
##  3rd Qu.:61038   3rd Qu.:11.500   3rd Qu.:96.50         3rd Qu.: 8.300
##  Max.   :88479   Max.   :17.400   Max.   :98.10         Max.   :13.300
##  perc_ed_gt_hs   perc_ed_gt_bach perc_ed_gt_grad     county
##  Min.   :78.40   Min.   :12.70   Min.   : 2.600   Length:8893
##  1st Qu.:90.40   1st Qu.:17.50   1st Qu.: 4.500   Class :character
##  Median :92.40   Median :19.20   Median : 5.300   Mode  :character
##  Mean   :91.57   Mean   :20.26   Mean   : 5.692
##  3rd Qu.:93.30   3rd Qu.:22.20   3rd Qu.: 6.200
##  Max.   :96.40   Max.   :50.40   Max.   :15.400
##    county_num      area_sq_mi         lat             long
##  Min.   : 1.00   Min.   :380.5   Min.   :40.65   Min.   :-96.22
##  1st Qu.:24.00   1st Qu.:473.2   1st Qu.:41.33   1st Qu.:-94.90
##  Median :48.00   Median :569.3   Median :42.07   Median :-93.74
##  Mean   :48.66   Mean   :555.9   Mean   :42.06   Mean   :-93.63
##  3rd Qu.:72.00   3rd Qu.:580.4   3rd Qu.:42.78   3rd Qu.:-92.33
##  Max.   :99.00   Max.   :972.7   Max.   :43.39   Max.   :-90.57
##    time_index
##  Min.   :  0.00
##  1st Qu.: 27.00
##  Median : 54.00
##  Mean   : 54.27
##  3rd Qu.: 82.00
##  Max.   :110.00
```

```
rm(i,full_list,outlier_switch,outliers)
```

Normalizing

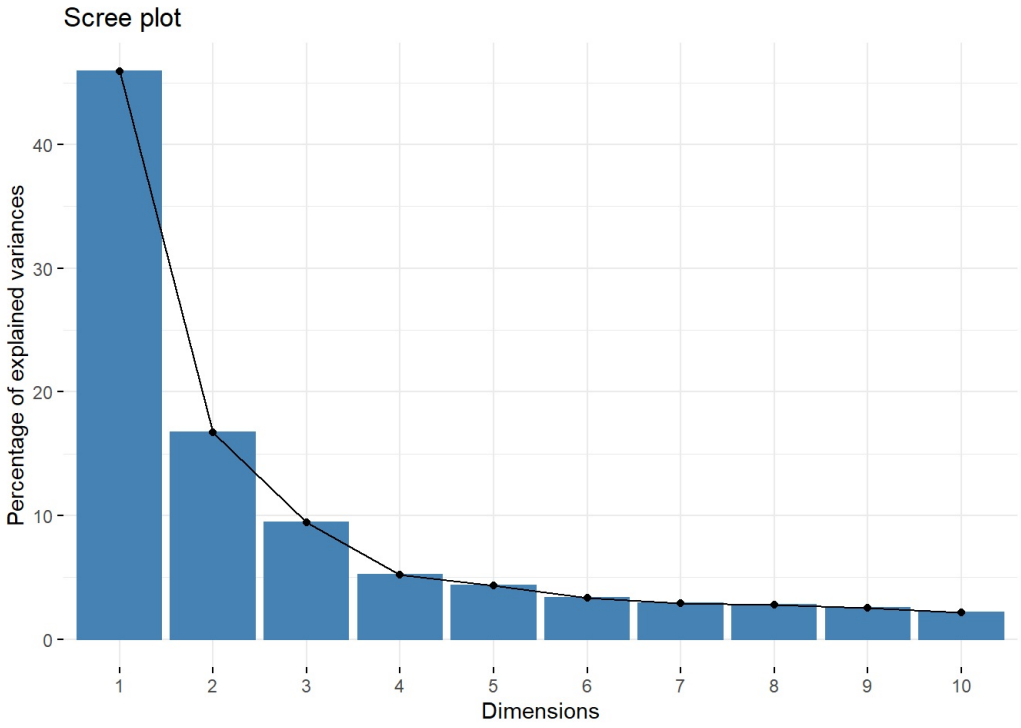
We normalize the data to better visualize and cluster it.

```
Numeric_Dataset <- dataset[ , unlist(lapply(dataset, is.numeric))]
i <-1
for(i in seq.int(ncol(Numeric_Dataset))) {
  max <- max(Numeric_Dataset[,i])
  Numeric_Dataset[,i] <- Numeric_Dataset[,i] / max
}
rm(max)
```

PCA

We applied Principal Component analysis to see the variance among the attributes.

```
discards <- c("county_num","time_index")
pca <- prcomp(Numeric_Dataset[ , !(names(Numeric_Dataset) %in% discards)],scale=FALSE)
fviz_eig(pca)
```

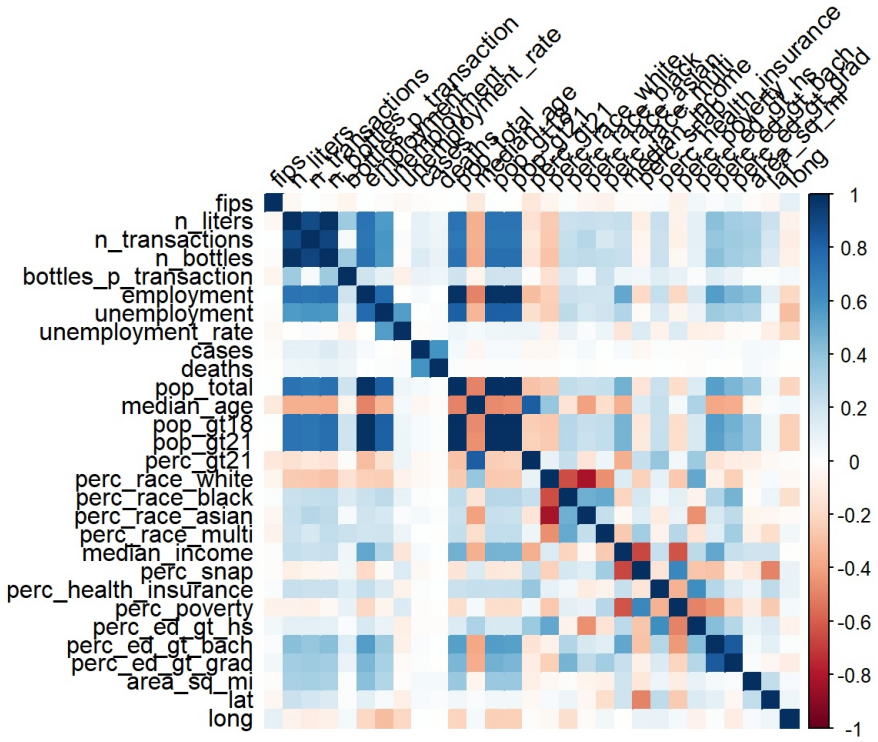



```
summary(pca)
```

```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     0.4099 0.2472 0.18568 0.13756 0.12566 0.11076 0.10290
## Proportion of Variance 0.4593 0.1671 0.09424 0.05172 0.04316 0.03353 0.02894
## Cumulative Proportion  0.4593 0.6263 0.72059 0.77232 0.81548 0.84901 0.87796
##                            PC8     PC9    PC10    PC11    PC12    PC13    PC14
## Standard deviation     0.10019 0.09631 0.08871 0.07495 0.05167 0.04758 0.04065
## Proportion of Variance 0.02744 0.02536 0.02151 0.01536 0.00730 0.00619 0.00452
## Cumulative Proportion  0.90539 0.93075 0.95226 0.96762 0.97491 0.98110 0.98562
##                           PC15    PC16    PC17    PC18    PC19    PC20    PC21
## Standard deviation     0.03395 0.02863 0.02561 0.02474 0.02365 0.02187 0.01707
## Proportion of Variance 0.00315 0.00224 0.00179 0.00167 0.00153 0.00131 0.00080
## Cumulative Proportion  0.98877 0.99101 0.99280 0.99448 0.99600 0.99731 0.99811
##                           PC22    PC23    PC24    PC25     PC26     PC27
## Standard deviation     0.01487 0.01195 0.01085 0.01068 0.009061 0.002732
## Proportion of Variance 0.00060 0.00039 0.00032 0.00031 0.000220 0.000020
## Cumulative Proportion  0.99871 0.99910 0.99942 0.99974 0.999960 0.999980
##                           PC28     PC29
## Standard deviation     0.002053 0.001582
## Proportion of Variance 0.000010 0.000010
## Cumulative Proportion  0.999990 1.000000
```

The elbow rule dictates we take 7 Principal Components.

```
cors <- cor(Numeric_Dataset[ , !(names(Numeric_Dataset) %in% discards)])
corrplot(cors
         ,method="color"
         ,tl.col="black"
         ,tl.srt=45
         ,na.label = " ")
```




```
rm(cors)
```

```
# We looked at the correlation plot to see how the attributes correlate.
res.pca <-PCA(Numeric_Dataset[ , !(names(Numeric_Dataset) %in% discards)], scale.unit = TRUE, ncp = 20, graph = F
ALSE)
res.pca$var$contrib
```

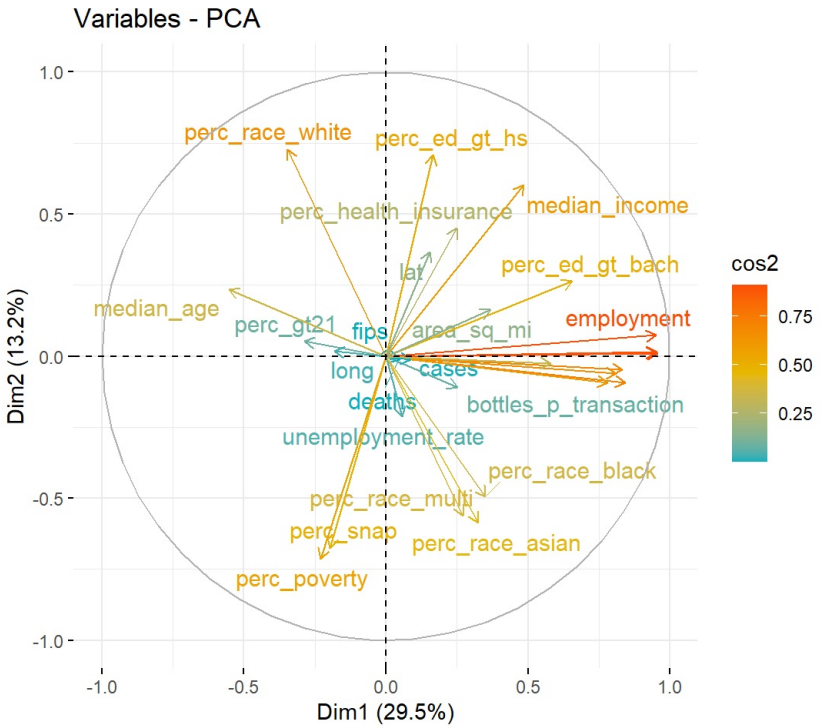
```
##                            Dim.1        Dim.2        Dim.3       Dim.4
## fips                  0.001683442  0.019046841 6.660660e-01  0.81334597
## n_liters              8.100649702  0.058529299 7.204742e-01  1.71738294
## n_transactions        7.673538168  0.096505419 5.628919e-01  1.74233484
## n_bottles             8.296459685  0.224630146 8.389877e-01  1.67692780
## bottles_p_transaction 0.737743533  0.315510919 1.231675e+00  0.02931007
## employment           10.568707382  0.140697293 7.299589e-02  0.88122938
## unemployment          7.092548750  0.224954105 4.106457e-01  5.34182928
## unemployment_rate     0.038528846  1.186092154 1.089859e+00  6.17850243
## cases                 0.085064089  0.003271901 1.062343e-02  0.45907449
## deaths                0.026357553  0.003130539 6.958784e-04  0.87656913
## pop_total            10.592538822  0.006007585 5.959258e-03  1.09164684
## median_age            3.567536122  1.427809336 1.830961e+01  1.39341857
## pop_gt18             10.681514388  0.002015014 3.499707e-02  0.94005816
## pop_gt21             10.575036583  0.004149077 7.498643e-02  1.12547317
## perc_gt21             0.959643042  0.070393417 3.032608e+01  0.16211626
## perc_race_white       1.412790218 13.789582948 2.694271e-01  4.58922715
## perc_race_black       1.434448403  6.389314538 4.830496e+00 10.48498946
## perc_race_asian       1.237289733  8.951791024 1.353230e+00  6.40378519
## perc_race_multi       0.857034289  8.210026625 3.038546e+00  2.73738834
## median_income         2.751513261  9.449635973 6.152277e+00  0.74941226
## perc_snap             0.451436101 11.850620438 5.343427e+00  1.48228865
## perc_health_insurance 0.729898956  5.277853573 1.212632e+01  1.39847715
## perc_poverty          0.616093417 13.235929441 5.186296e-01  5.38030218
## perc_ed_gt_hs         0.322922961 13.040077739 1.113811e+01  1.19881203
## perc_ed_gt_bach       5.016744679  1.802510291 6.488735e-02 14.59187702
## perc_ed_gt_grad       3.919624213  0.017784135 6.066113e-01 17.25199908
## area_sq_mi            1.588236083  0.691861129 5.633770e-03  7.05493527
## lat                   0.278857394  3.501723717 2.440975e-02  0.09420233
## long                  0.385560184  0.008545384 1.714400e-01  2.15308457
##                            Dim.5        Dim.6       Dim.7        Dim.8
## fips                  0.034227577 2.876045e-02  3.34517218 56.721265503
## n_liters              3.967007791 3.074622e+00  0.11705294  0.150289574
## n_transactions        1.815269783 1.329753e+00  0.39241313  2.422074384
## n_bottles             4.220763661 2.117870e+00  0.30997414  0.040828928
```

```
## bottles_p_transaction  4.941616320 1.599724e+00  7.04698041 20.790897492
## employment             0.378886031 1.457835e-01  0.23732428  0.013250778
## unemployment           5.576771623 2.902848e+00  1.30798317  0.460124708
## unemployment_rate      9.100118672 1.163923e+01  6.00441396  1.298895084
## cases                 23.749540352 2.376835e+01  0.38729350  0.101092109
## deaths                20.981057195 2.679881e+01  0.36347859  0.437482746
## pop_total              0.806817783 3.171816e-04  0.40908628  0.005705249
## median_age             0.083861048 9.715139e-02  1.29017763  0.023552475
## pop_gt18               0.858616411 5.598075e-03  0.31650203  0.002135385
## pop_gt21               0.855284831 2.817345e-03  0.26215736  0.012480552
## perc_gt21              0.029730672 3.012950e-05  2.61092161  0.011076954
## perc_race_white        1.150518861 3.968871e-01  7.29224682  0.488470867
## perc_race_black        0.003873524 5.912146e-01  6.39565391  0.011055888
## perc_race_asian        0.689384508 2.396900e-03 10.24876648  0.015849883
## perc_race_multi        0.023250251 3.231244e-03  3.09575423  1.072443073
## median_income          0.094101108 2.267935e-01  0.03391542  1.663414173
## perc_snap              0.329434129 1.382032e-01  8.44728227  1.594724541
## perc_health_insurance  0.861454614 7.512275e-01  0.30330892  1.249005320
## perc_poverty           0.072789494 1.028507e+00  2.43395456  0.039262995
## perc_ed_gt_hs          0.452761522 4.601456e-01  1.88682772  0.446288352
## perc_ed_gt_bach        0.470546965 1.323523e+00  1.96917155  0.032346438
## perc_ed_gt_grad        1.492173442 2.824794e+00  2.58323286  0.137586375
## area_sq_mi             1.449847413 3.532375e+00  2.40650846  3.684117404
## lat                    6.832371183 3.697942e+00 24.32303567  0.005964490
## long                   8.677923235 1.151110e+01  4.17940991  7.068318279
##                              Dim.9       Dim.10       Dim.11      Dim.12
## fips                  9.328732e+00 20.696724301  1.546255996  0.99111670
## n_liters              1.656880e-01  2.195076871  0.833390340  7.78868386
## n_transactions        7.290951e+00  0.707855268  2.504766279  8.51484358
## n_bottles             3.866748e-01  2.713725221  1.278106731  5.40447438
## bottles_p_transaction 3.604296e+01  5.219889763  0.393283024  2.18040493
## employment            1.772300e-01  0.009158947  0.126928706  2.82186210
## unemployment          5.081604e+00  1.468252635  0.414646582  0.06222635
## unemployment_rate     2.072203e+01  5.469042367  1.804561453  9.60956422
## cases                 2.231000e-01  0.085265909  0.008119091  0.30559140
## deaths                2.916691e-02  0.346198083  0.058557729  0.28682663
## pop_total             2.306036e-02  0.043647025  0.001555278  2.73312493
## median_age            7.243693e-02  0.010987533  0.019532052  1.88975344
## pop_gt18              6.294707e-02  0.007006395  0.016093929  2.57303242
## pop_gt21              7.682426e-02  0.013478900  0.002184961  2.96362042
## perc_gt21             8.859770e-01  0.006747835  0.761456787  2.97292806
## perc_race_white       1.008445e+00  0.006678826  1.007714397  5.36193362
## perc_race_black       1.149885e-01  0.983768395  8.727586035  0.29315040
## perc_race_asian       8.066202e-01  0.090029117 13.336593506  2.85489686
## perc_race_multi       7.766242e-01  4.262165398  8.736015791  0.36226358
## median_income         1.472843e+00  1.643347702  2.611983146  4.81446584
## perc_snap             1.435813e+00  1.207035536  1.226067692  0.35184355
## perc_health_insurance 2.276025e+00  0.019501755 11.821622891  2.25124459
## perc_poverty          6.643431e-05  0.012398989  1.494608918  1.58061629
## perc_ed_gt_hs         1.193664e-01  0.132528750  0.498537649  0.40798314
## perc_ed_gt_bach       6.664980e-02  2.464839916  0.026534129  3.44476313
## perc_ed_gt_grad       1.719052e+00  1.374044620  3.338678052  3.76374576
## area_sq_mi            8.611725e-01 12.111120653 23.193360856 11.86108096
## lat                   3.563224e+00  0.046390699  9.882906200 10.74155038
## long                  5.209725e+00 36.653092590  4.328351801  0.81240848
##                             Dim.13       Dim.14      Dim.15       Dim.16
## fips                  0.013198500 1.272728e+00  0.05167235 7.220356e-01
## n_liters              0.967450424 3.531500e+00  0.01207475 4.817635e-01
## n_transactions        0.001360090 2.024428e+00  0.01967785 3.792353e-01
## n_bottles             0.703716805 3.658238e+00  0.03156963 3.530492e-01
## bottles_p_transaction 9.213287681 1.163580e-01  0.01784433 3.425380e-01
## employment            1.160231513 8.713949e-01  0.32975199 1.222665e+00
## unemployment          0.146156153 8.976179e-02  0.06249874 6.280184e-01
## unemployment_rate     2.653153113 1.331411e+00  0.81146085 1.030157e+00
## cases                 0.295007099 4.025828e-02 40.75046118 9.471082e+00
## deaths                0.117204303 2.265800e-01 42.85302563 6.515327e+00
## pop_total             1.494115912 4.815191e-01  0.38283994 1.984499e+00
## median_age            0.474519644 5.375795e+00  1.72466442 8.368609e+00
## pop_gt18              1.207641647 3.797332e-01  0.37380552 1.943334e+00
## pop_gt21              1.456302570 2.231425e-01  0.42842818 2.156624e+00
## perc_gt21             0.048907315 5.365470e+00  1.12989694 6.670822e+00
## perc_race_white       0.003107317 7.648964e-02  0.01923126 1.170647e-01
## perc_race_black       0.006532792 2.568900e-02  0.29487320 1.289684e+00
## perc_race_asian       3.738490062 5.650731e-04  0.07904095 8.989442e-01
## perc_race_multi      35.252034926 9.208579e+00  1.18076774 6.895529e+00
## median_income         0.196128564 1.566102e+00  0.36723941 2.440989e+00
## perc_snap             1.430131493 8.546977e+00  0.11619129 8.999227e-01
## perc_health_insurance 3.585451905 2.271062e+01  2.70611932 1.191529e+01
## perc_poverty          0.324058882 1.833245e+01  0.99035339 1.082496e+01
```

```
## perc_ed_gt_hs          0.087404465 4.274993e-01  0.82699990 3.221155e+00
## perc_ed_gt_bach        7.282850855 7.623964e-01  0.64184712 3.484225e+00
## perc_ed_gt_grad        8.467017237 3.787434e-01  0.00972687 2.399533e-04
## area_sq_mi            16.352789535 3.723087e-01  2.30430181 8.622134e+00
## lat                    3.110499064 1.139700e+01  0.50230940 2.038069e+00
## long                   0.211250134 1.206262e+00  0.98132605 5.082030e+00
##                             Dim.17       Dim.18       Dim.19       Dim.20
## fips                   0.477557921 1.868019e+00  0.246731265 3.830522e-01
## n_liters               0.065180363 4.445624e-02  0.247067564 1.514785e-01
## n_transactions         0.471888560 9.302438e-01  2.130467585 3.754492e-01
## n_bottles              0.182210708 3.694428e-01  0.476305409 2.141118e-01
## bottles_p_transaction  0.553761750 3.174241e-01  0.116414502 4.036715e-02
## employment             0.256213055 4.566928e-01  0.898220958 7.144805e-01
## unemployment           0.599703363 3.016177e-02  0.560070312 2.823150e-01
## unemployment_rate      0.270806509 7.595805e-02  0.058312088 1.453310e-01
## cases                  0.151553344 2.121142e-02  0.009048928 1.308087e-02
## deaths                 0.032681339 2.958314e-04  0.026388350 7.598305e-05
## pop_total              0.887745289 1.435771e-01  1.138474400 4.426451e-01
## median_age             0.005757688 1.047745e+01  0.510145025 1.597418e+00
## pop_gt18               0.740596918 6.964391e-02  0.759995449 5.630754e-01
## pop_gt21               0.832969467 4.392155e-02  0.745374395 2.607890e-01
## perc_gt21              1.884964908 4.170764e-02  0.014189636 4.267814e-01
## perc_race_white        2.578031790 1.874869e-01  0.372879802 4.112791e-02
## perc_race_black        6.021287780 3.734995e+01  3.598334151 2.129240e-02
## perc_race_asian        0.030228307 1.038170e+01 10.767424955 6.306057e+00
## perc_race_multi        2.777236284 6.097261e+00  0.096373069 2.350081e+00
## median_income          7.064127316 3.245044e-01 23.816203613 2.509619e+01
## perc_snap             22.478176553 4.219043e+00 15.012432423 1.022216e+01
## perc_health_insurance  5.850567186 4.498884e-01  3.825130481 6.693671e+00
## perc_poverty          34.130635398 1.139740e+00  1.627732503 4.244820e+00
## perc_ed_gt_hs          0.313584666 2.817467e+00 28.340746705 2.436864e+01
## perc_ed_gt_bach        0.008729502 1.147336e+00  0.629349884 1.104745e+00
## perc_ed_gt_grad        3.925049162 6.473175e+00  0.588406160 1.084763e+01
## area_sq_mi             1.591070661 5.099849e-04  0.178257802 1.091938e-01
## lat                    3.434468464 1.132100e+01  0.314865477 2.154888e+00
## long                   2.383215750 3.200729e+00  2.894657109 8.290569e-01
```

```
# Color by cos2 values: quality on the factor map
fviz_pca_var(res.pca, col.var = "cos2",
             gradient.cols = c("#00AFBB", "#E7B800", "#FC4E07"),
             repel = TRUE
             )
```

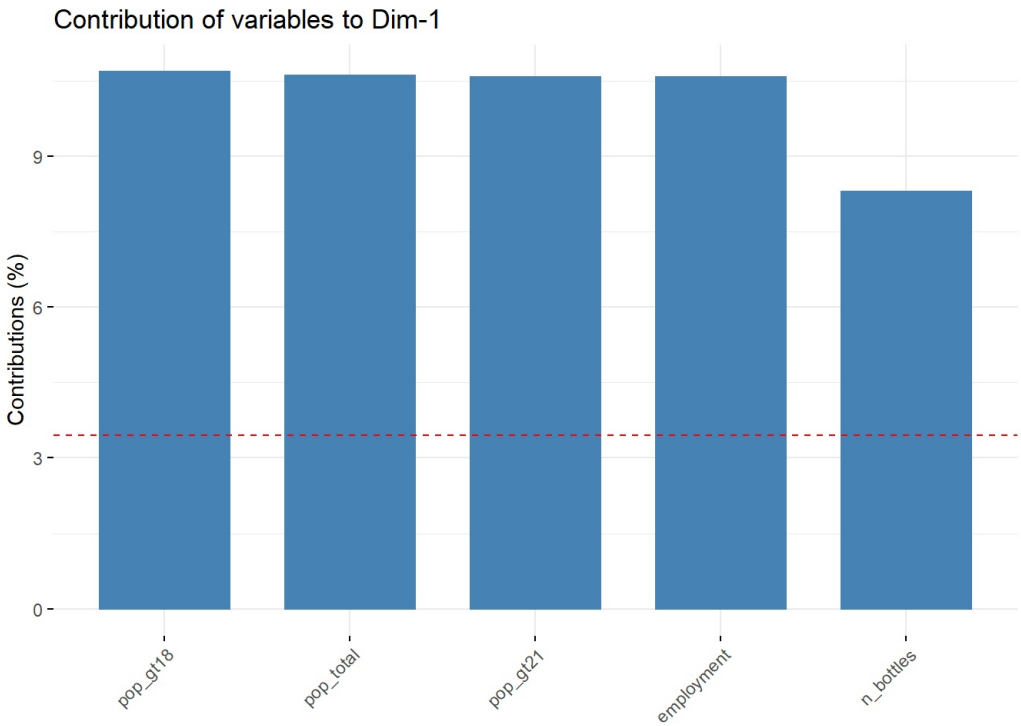
```
## Warning: ggrepel: 8 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

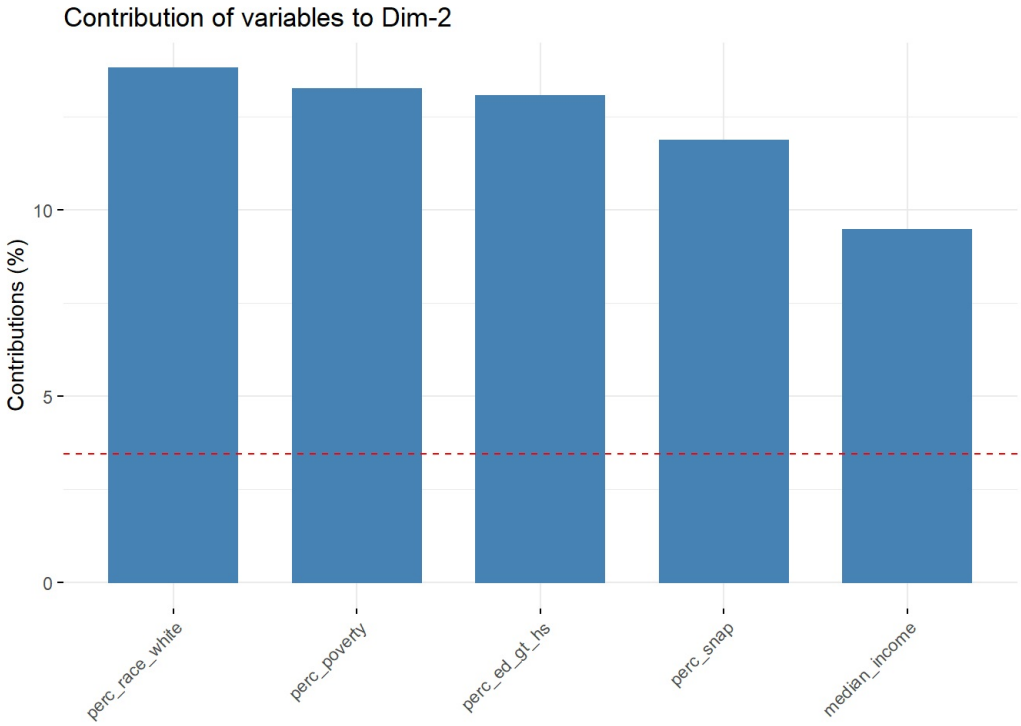


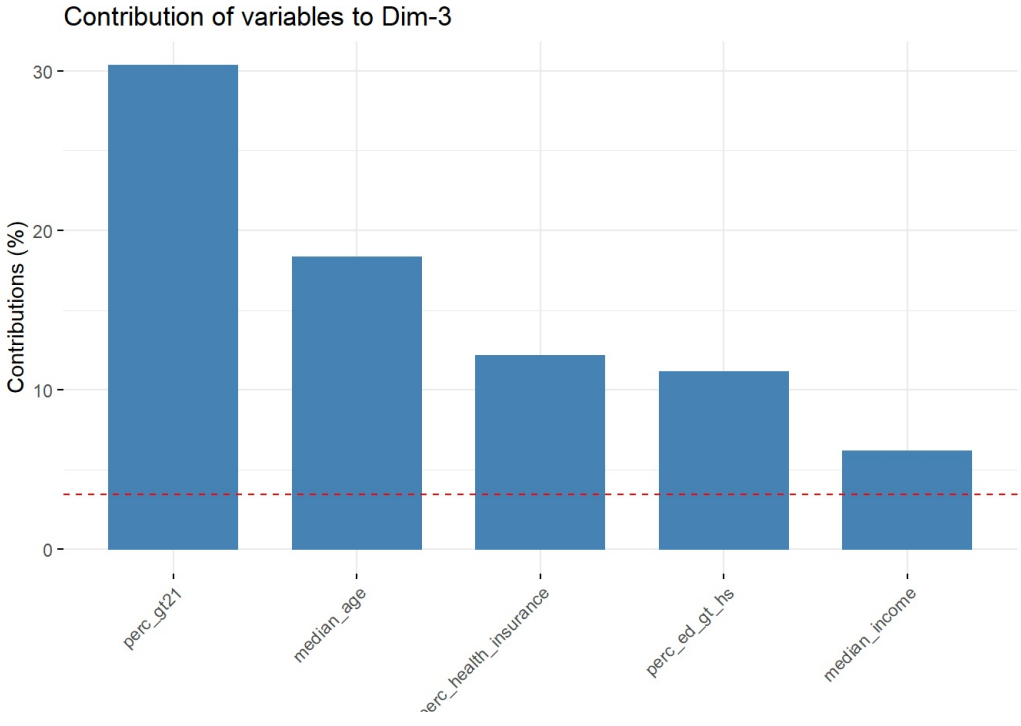

```
summary(pca)
```

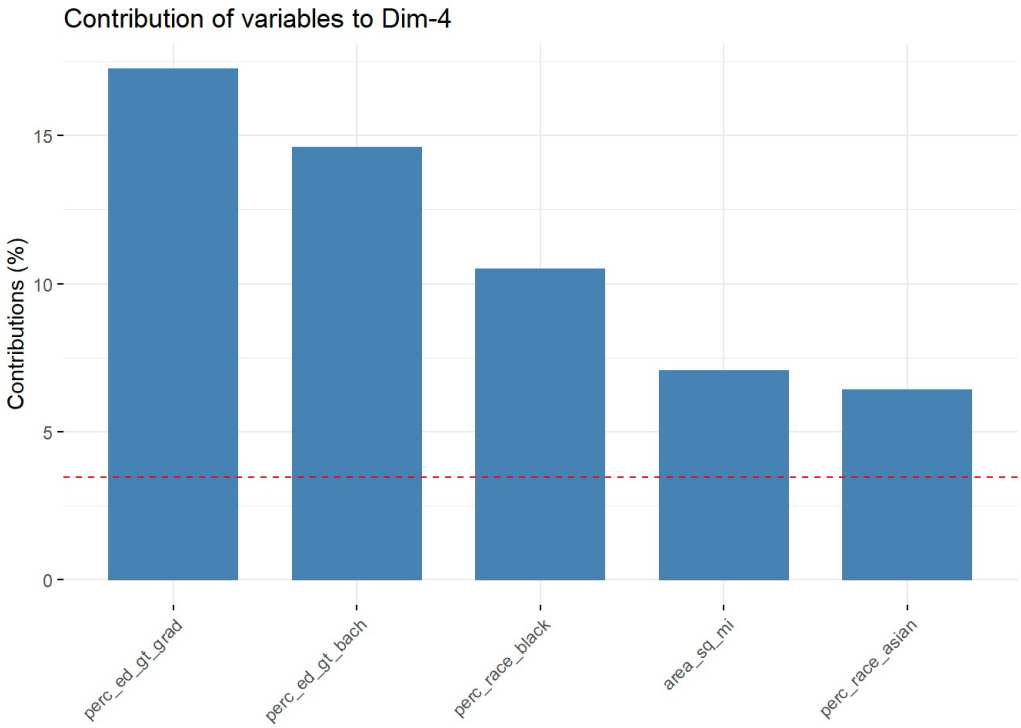
```
## Importance of components:
##                           PC1    PC2     PC3     PC4     PC5     PC6     PC7
## Standard deviation     0.4099 0.2472 0.18568 0.13756 0.12566 0.11076 0.10290
## Proportion of Variance 0.4593 0.1671 0.09424 0.05172 0.04316 0.03353 0.02894
## Cumulative Proportion  0.4593 0.6263 0.72059 0.77232 0.81548 0.84901 0.87796
##                            PC8     PC9    PC10    PC11    PC12    PC13    PC14
## Standard deviation     0.10019 0.09631 0.08871 0.07495 0.05167 0.04758 0.04065
## Proportion of Variance 0.02744 0.02536 0.02151 0.01536 0.00730 0.00619 0.00452
## Cumulative Proportion  0.90539 0.93075 0.95226 0.96762 0.97491 0.98110 0.98562
##                           PC15    PC16    PC17    PC18    PC19    PC20    PC21
## Standard deviation     0.03395 0.02863 0.02561 0.02474 0.02365 0.02187 0.01707
## Proportion of Variance 0.00315 0.00224 0.00179 0.00167 0.00153 0.00131 0.00080
## Cumulative Proportion  0.98877 0.99101 0.99280 0.99448 0.99600 0.99731 0.99811
##                           PC22    PC23    PC24    PC25     PC26     PC27
## Standard deviation     0.01487 0.01195 0.01085 0.01068 0.009061 0.002732
## Proportion of Variance 0.00060 0.00039 0.00032 0.00031 0.000220 0.000020
## Cumulative Proportion  0.99871 0.99910 0.99942 0.99974 0.999960 0.999980
##                           PC28     PC29
## Standard deviation     0.002053 0.001582
## Proportion of Variance 0.000010 0.000010
## Cumulative Proportion  0.999990 1.000000
```

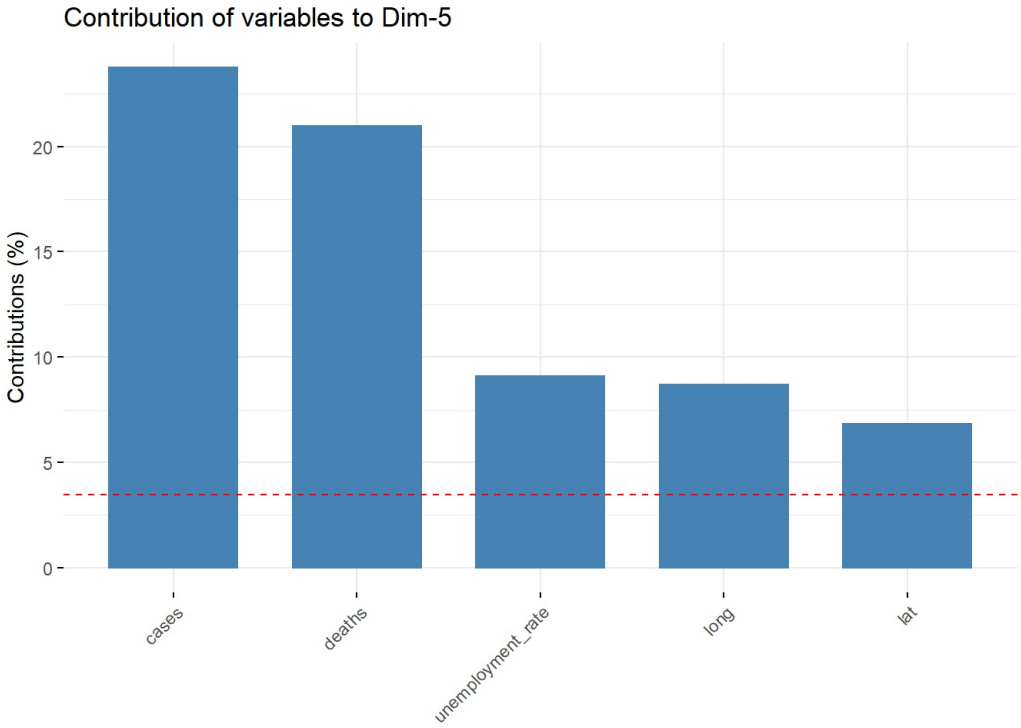
```
for(i in seq.int(15)){
  print(fviz_contrib(res.pca, choice = "var", axes = i, top = 5))
}
```

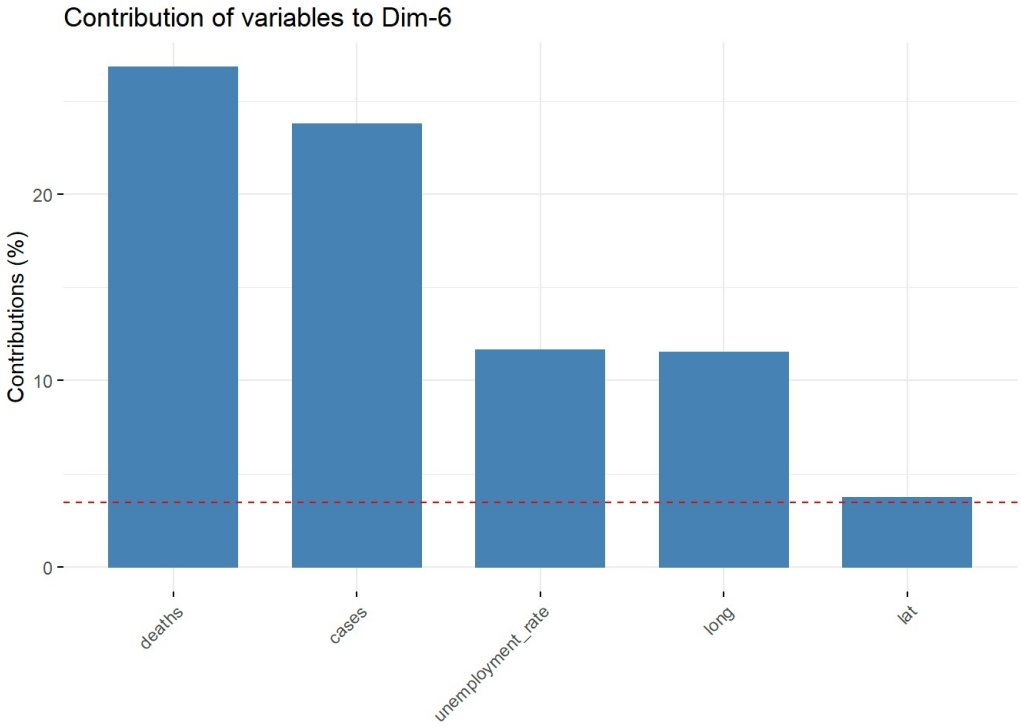

Contribution of variables to Dim-1
Contributions (%)
0
3
6
9
pop_gt18
pop_total
pop_gt21
employment
n_bottles

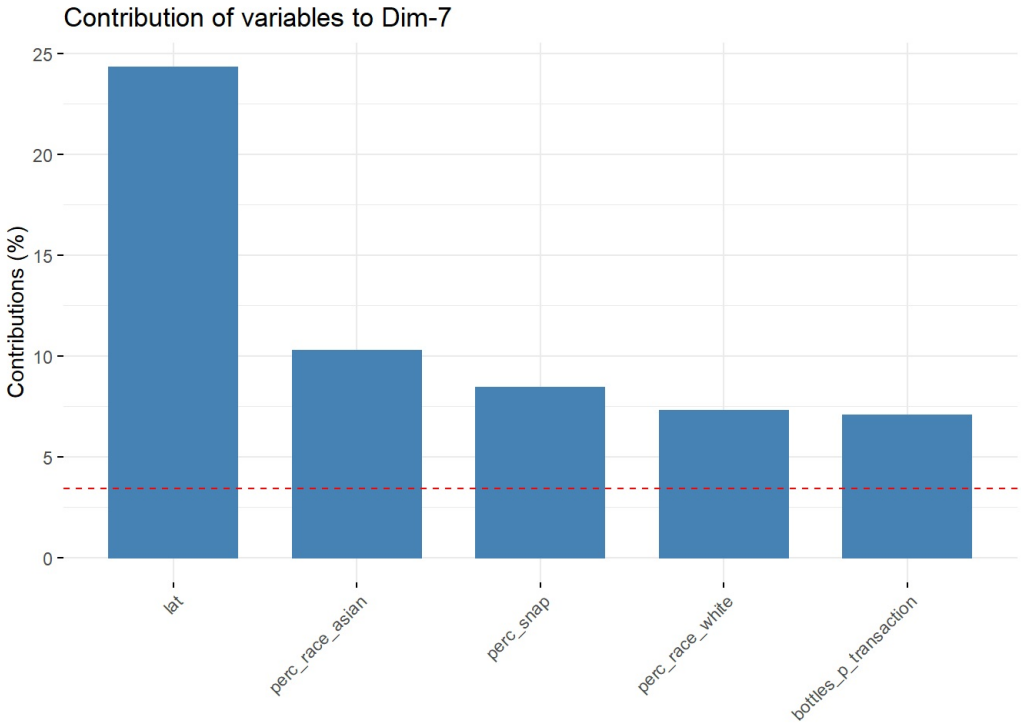

Contribution of variables to Dim-2
Contributions (%)
0
5
10
perc_race_white
perc_poverty
perc_ed_gt_hs
perc_snap
median_income

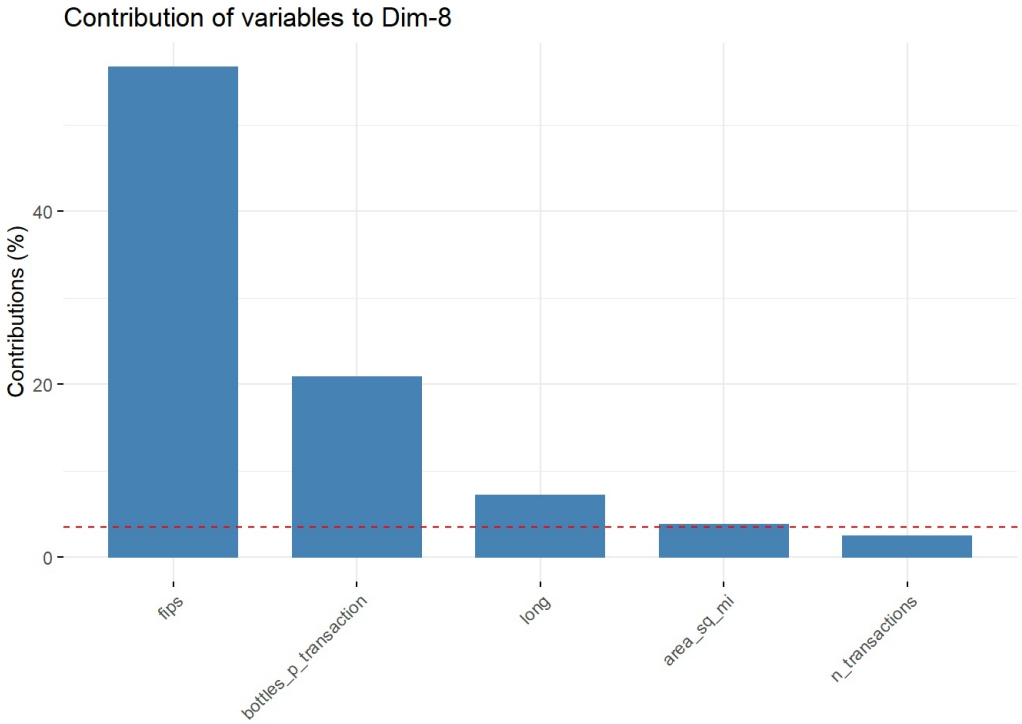

Contribution of variables to Dim-3
Contributions (%)
0
10
20
30
perc_gt21
median_age
perc_health_insurance
perc_ed_gt_hs
median_income

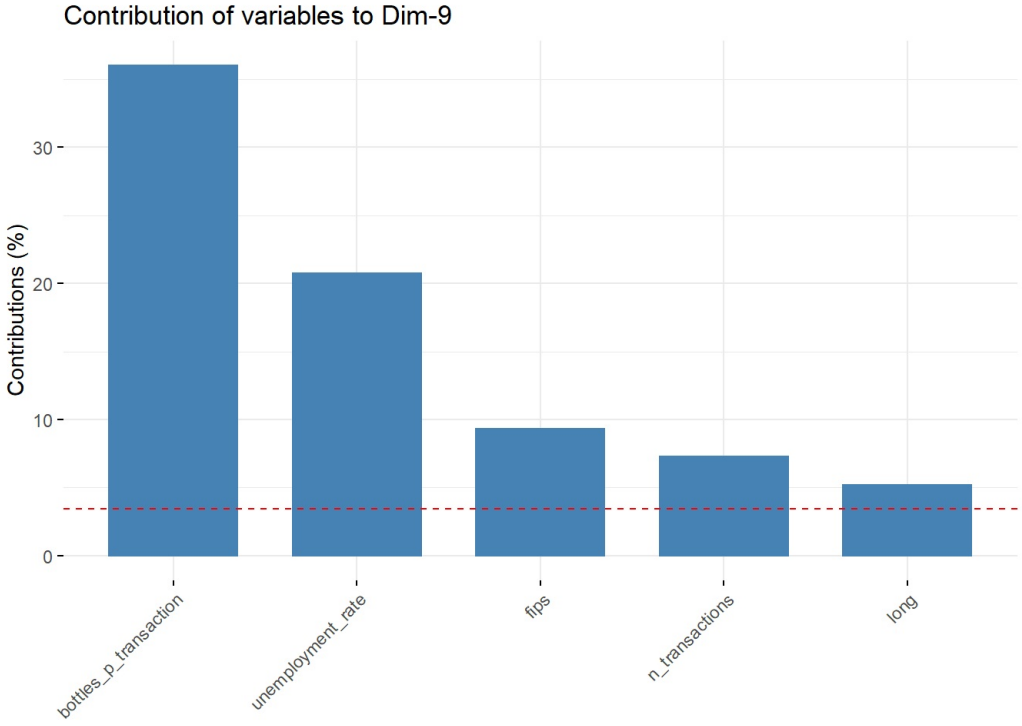

Contribution of variables to Dim-4
Contributions (%)
0
5
10
15
perc_ed_gt_grad
perc_ed_gt_bach
perc_race_black
area_sq_mi
perc_race_asian

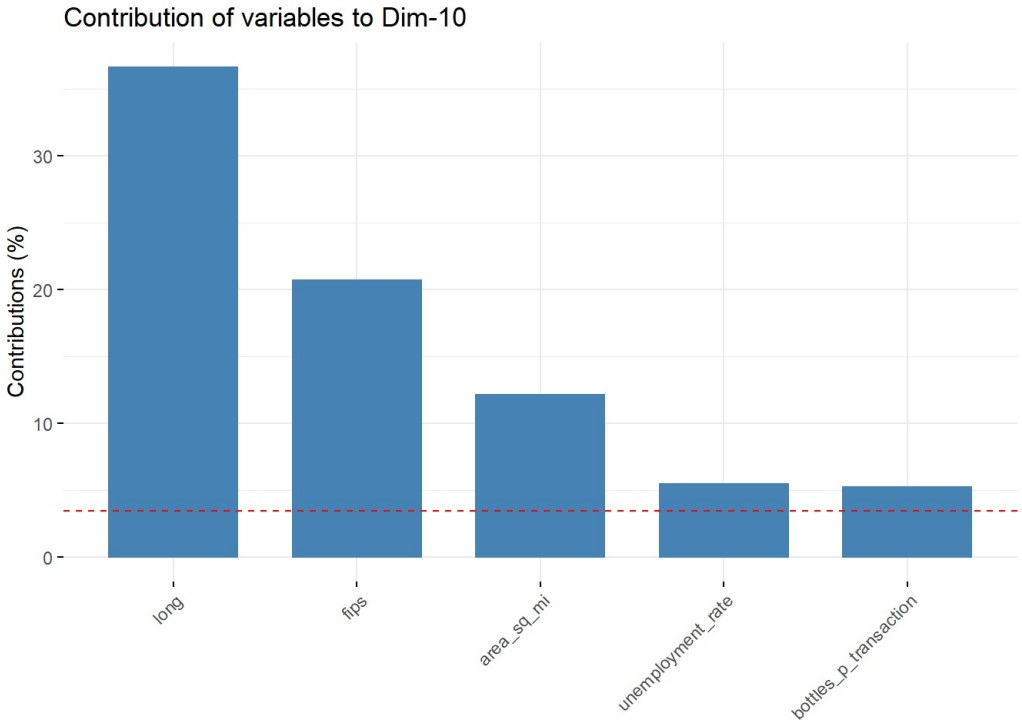

Contribution of variables to Dim-5
Contributions (%)
0
5
10
15
20
cases
deaths
unemployment_rate
long
lat

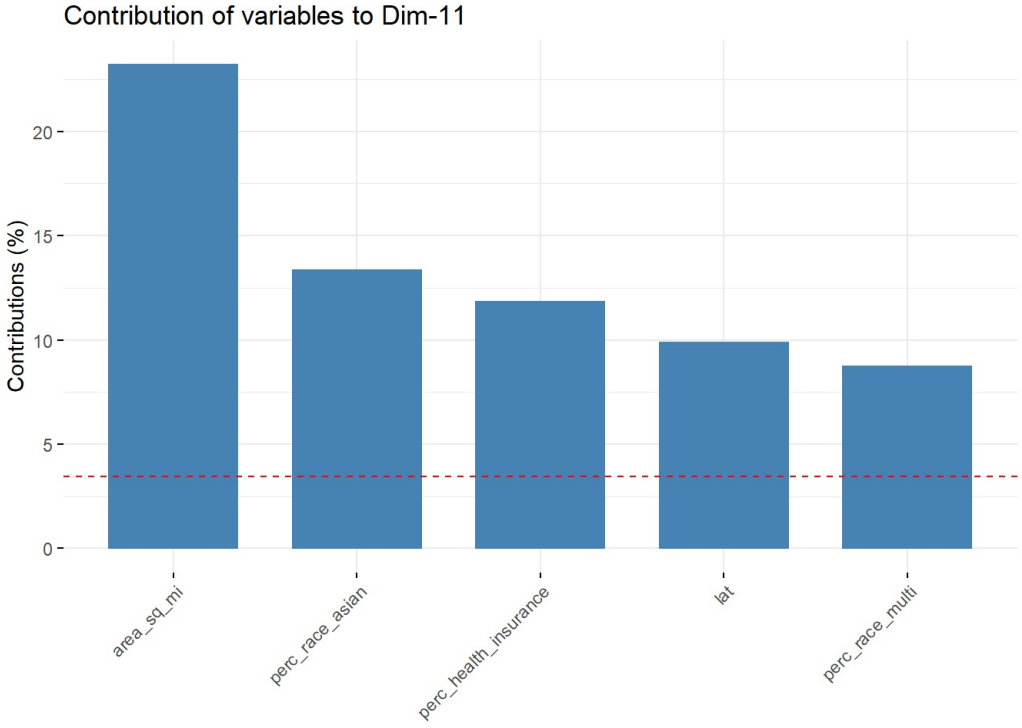

Contribution of variables to Dim-6
Contributions (%)
0
10
20
deaths
cases
unemployment_rate
long
lat

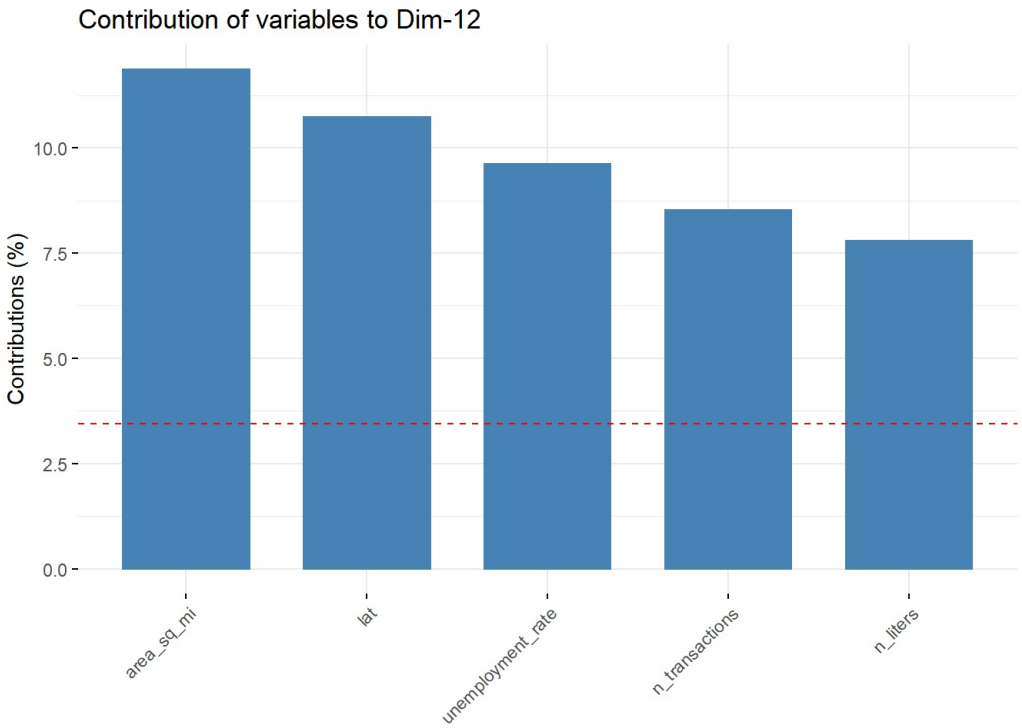

Contribution of variables to Dim-7
Contributions (%)
0
5
10
15
20
25
lat
perc_race_asian
perc_snap
perc_race_white
bottles_p_transaction

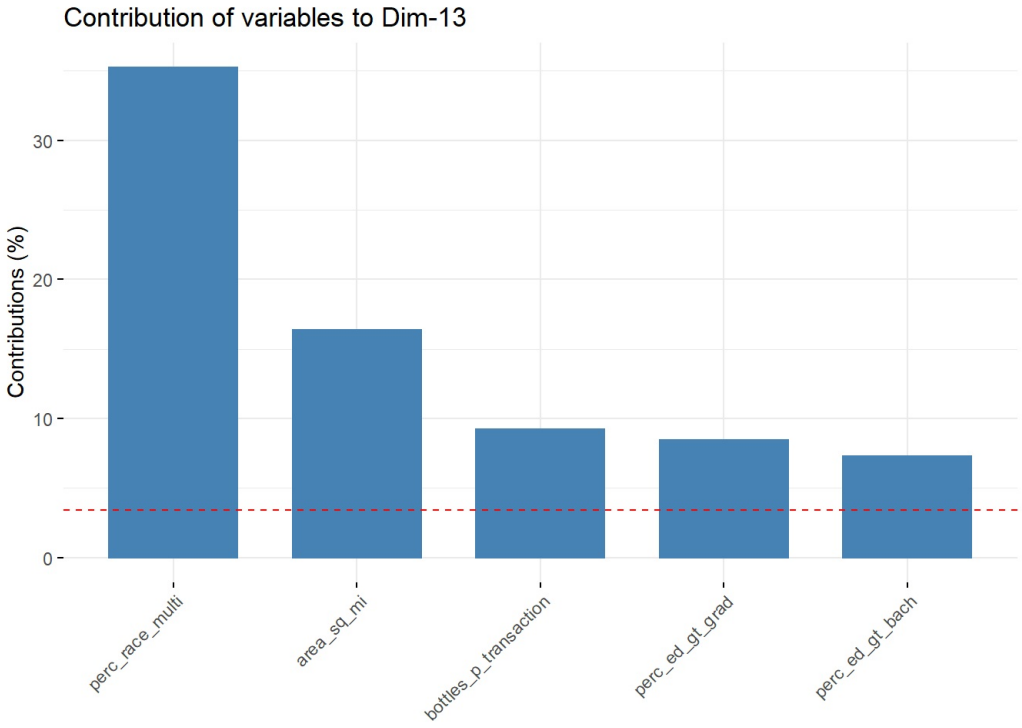

Contribution of variables to Dim-8
Contributions (%)
40
20
0
fips
bottles_p_transaction
long
area_sq_mi
n_transactions

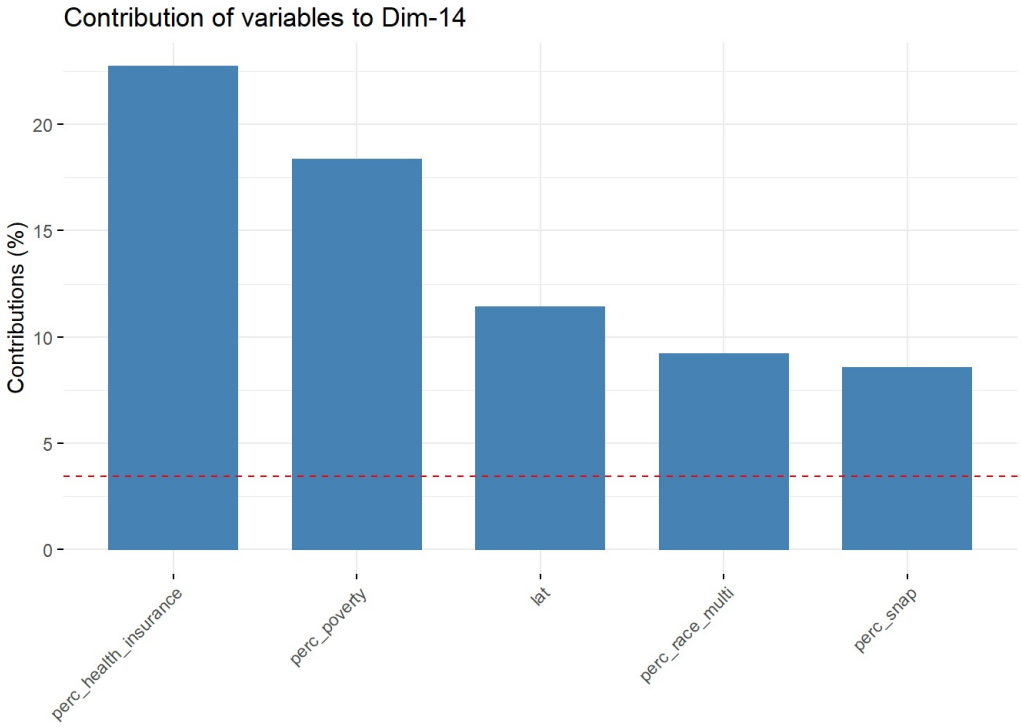

Contribution of variables to Dim-9
Contributions (%)
30
20
10
0
bottles_p_transaction
unemployment_rate
fips
n_transactions
long

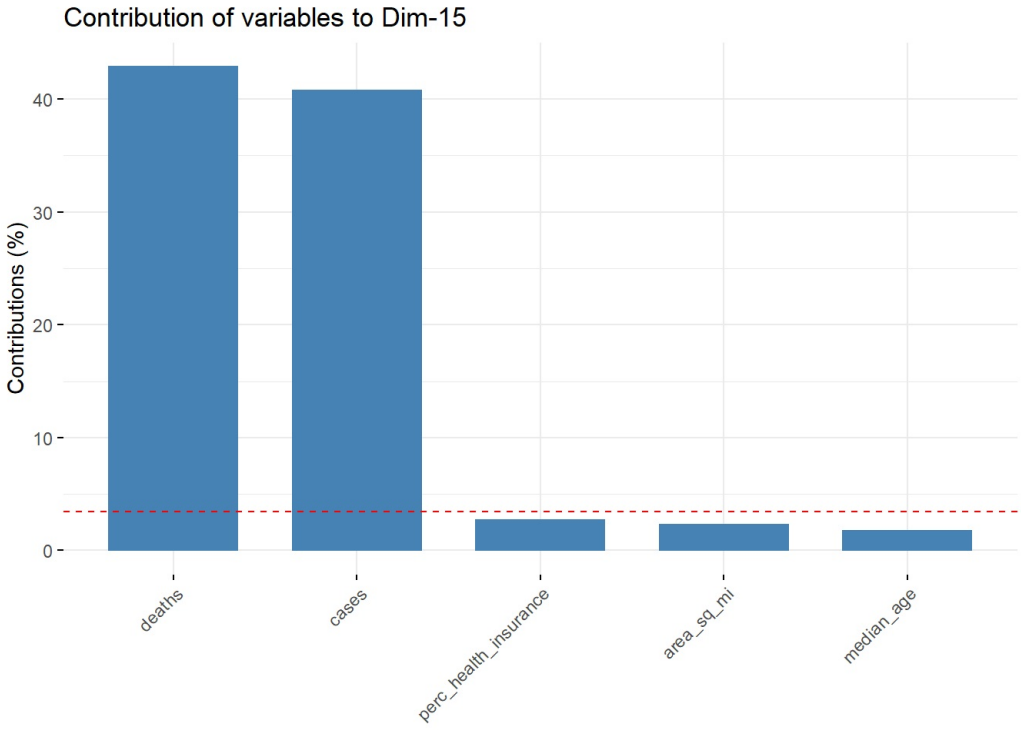

Contribution of variables to Dim-10
Contributions (%)
30
20
10
0
long
fips
area_sq_mi
unemployment_rate
bottles_p_transaction


Contribution of variables to Dim-11
Contributions (%)
20
15
10
5
0
area_sq_mi
perc_race_asian
perc_health_insurance
lat
perc_race_multi


Contribution of variables to Dim-12
Contributions (%)
10.0
7.5
5.0
2.5
0.0
area_sq_mi
lat
unemployment_rate
n_transactions
n_liters


Contribution of variables to Dim-13
Contributions (%)
30
20
10
0
perc_race_multi
area_sq_mi
bottles_p_transaction
perc_ed_gt_grad
perc_ed_gt_bach






```
#[1] "pop_gt18"          "perc_poverty"      "perc_gt21"         "fips"
#[5] "perc_race_asian"   "area_sq_mi"        "unemployment_rate"
pcs <- c("n_bottles", "perc_poverty","perc_ed_gt_grad", "unemployment",  "perc_race_asian", "perc_snap", "area_sq
_mi", "deaths")
dataset_pca <- dataset[,pcs]
dataset_pca <- na.omit(dataset_pca)
rm(i,pca,discards,res.pca)
```

We take the most affecting attribute for each 7 of the Principal Components we have decided to pick.

```
index <- rowSums( Numeric_Dataset > 0) >= 1
Numeric_Dataset <- Numeric_Dataset[index,]

Numeric_Dataset <- Numeric_Dataset + log(1.1)
```

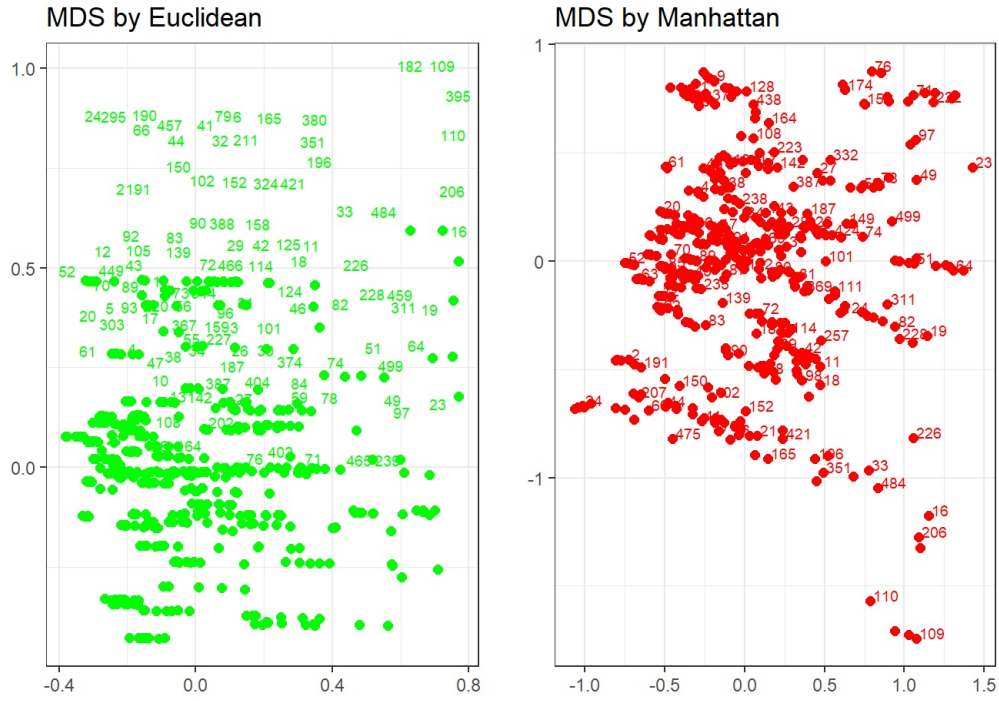
```
# Since we still have 8893 data points we sample it for easier processing
sample_data <- sample_n(Numeric_Dataset,500)
sample_data_pca <- sample_data[,pcs]
head(sample_data)
```

```
##       fips  n_liters n_transactions n_bottles bottles_p_transaction employment
## 1 1.095310 0.3094969      0.2888586 0.3302223             0.3871209  0.2579292
## 2 1.091143 0.2287621      0.3789123 0.2548557             0.2305673  0.1962626
## 3 1.092393 0.4906563      0.7229984 0.5766134             0.2796674  0.2800721
## 4 1.089684 0.2632445      0.2843783 0.2738713             0.3223771  0.2167388
## 5 1.089268 0.1816413      0.1822277 0.1875474             0.3504534  0.2362626
## 6 1.086559 0.5060590      0.5446829 0.5511047             0.3391741  0.2698340
##   unemployment unemployment_rate      cases     deaths pop_total median_age
## 1    0.2658984         0.3256473 0.09531018 0.09531018 0.2415690  0.9940705
## 2    0.2011925         0.3200293 0.09531018 0.09531018 0.1938532  1.0085333
## 3    0.2541337         0.2807034 0.17153830 0.22688913 0.2541524  0.9610127
## 4    0.2658984         0.3930630 0.09531018 0.09531018 0.2012356  0.9527482
## 5    0.2129572         0.2807034 0.09531018 0.09531018 0.2193300  1.0085333
## 6    0.2364867         0.2750855 0.09531018 0.09531018 0.2456104  1.0229961
##    pop_gt18  pop_gt21 perc_gt21 perc_race_white perc_race_black perc_race_asian
## 1 0.2495730 0.2505931  1.013221        1.037793       0.1508657      0.14481513
## 2 0.2003880 0.2001753  1.015708        1.095310       0.1091991      0.09531018
## 3 0.2634984 0.2637053  1.011977        1.062010       0.2758657      0.19432008
## 4 0.2050404 0.2046836  0.988345        1.085219       0.1369768      0.13491414
## 5 0.2293701 0.2291907  1.029390        1.070083       0.3175324      0.14481513
## 6 0.2575873 0.2580410  1.031877        1.074119       0.1230880      0.12501315
##   perc_race_multi median_income perc_snap perc_health_insurance perc_poverty
## 1       0.5163628     0.6980860 0.6182987              1.066768    0.6892951
## 2       0.2005733     0.7192435 0.8137010              1.061671    0.8095959
## 3       0.3584681     0.7022678 0.5723217              1.074923    0.5840320
## 4       0.3847839     0.7176047 0.4631263              1.053516    0.5915508
## 5       0.2532049     0.7933402 0.5033562              1.079000    0.7043327
## 6       0.4110997     0.6625295 0.9918619              1.069826    0.9374154
##   perc_ed_gt_hs perc_ed_gt_bach perc_ed_gt_grad county_num area_sq_mi      lat
## 1      1.053816       0.4365800       0.3550504  1.0953102  0.6919962 1.080177
## 2      1.050704       0.4326118       0.3225829  0.6912698  0.5380647 1.041005
## 3      1.050704       0.4683261       0.4005050  0.8124819  0.6844093 1.088261
## 4      1.038256       0.4088022       0.3745310  0.5498556  0.5818239 1.094750
## 5      1.063153       0.5675324       0.4134920  0.5094516  0.6823224 1.088069
## 6      1.008173       0.4941197       0.4524530  0.2468253  0.6753863 1.047930
##       long time_index
## 1 1.130200  0.3589465
## 2 1.125748  0.3680375
## 3 1.151077  1.0771284
## 4 1.114602  0.4225829
## 5 1.130299  0.2953102
## 6 1.143433  0.4680375
```

MDS

```
A <- sample_data_pca
# Evaluating Distance matrices

#weighted euclidean distance
dist.eu <- as.matrix(dist(A, method = "euclidean"))
mds.eu <- as.data.frame(cmdscale(dist.eu))
mds.eu$names <- rownames(mds.eu)

dist.man <- as.matrix(dist(A, method = "manhattan"))
mds.man <- as.data.frame(cmdscale(dist.man))
mds.man$names <- rownames(mds.man)
```

```
# Plotting the distance matrices on 2 dimensions using MDS.

plot3 <- ggplot(mds.eu, aes(V1, V2, label=names)) +
  geom_point(colour="green", size=2) +
  geom_text(colour="green", check_overlap = TRUE, size=2.5,
            hjust = "center", vjust = "bottom", nudge_x = 0, nudge_y = 0.4) +
  labs(x="", y="", title="MDS by Euclidean") + theme_bw()

plot4 <- ggplot(mds.man, aes(V1, V2, label=names)) +
  geom_point(colour="red", size=2) +
  geom_text(colour="red", check_overlap = TRUE, size=2.5,
            hjust = "left", vjust = "bottom", nudge_x = 0.02, nudge_y = 0) +
  labs(x="", y="", title="MDS by Manhattan") + theme_bw()

grid.arrange(plot3, plot4, ncol=2)
```




```
rm(A,plot2,plot3,plot4,plot5,dist.cos,dist.eu,dist.man,dist.max,mds.cos,mds.eu,mds.man,mds.max)
```

```
## Warning in rm(A, plot2, plot3, plot4, plot5, dist.cos, dist.eu, dist.man, :
## object 'plot2' not found
```

```
## Warning in rm(A, plot2, plot3, plot4, plot5, dist.cos, dist.eu, dist.man, :
## object 'plot5' not found
```

```
## Warning in rm(A, plot2, plot3, plot4, plot5, dist.cos, dist.eu, dist.man, :
## object 'dist.cos' not found
```

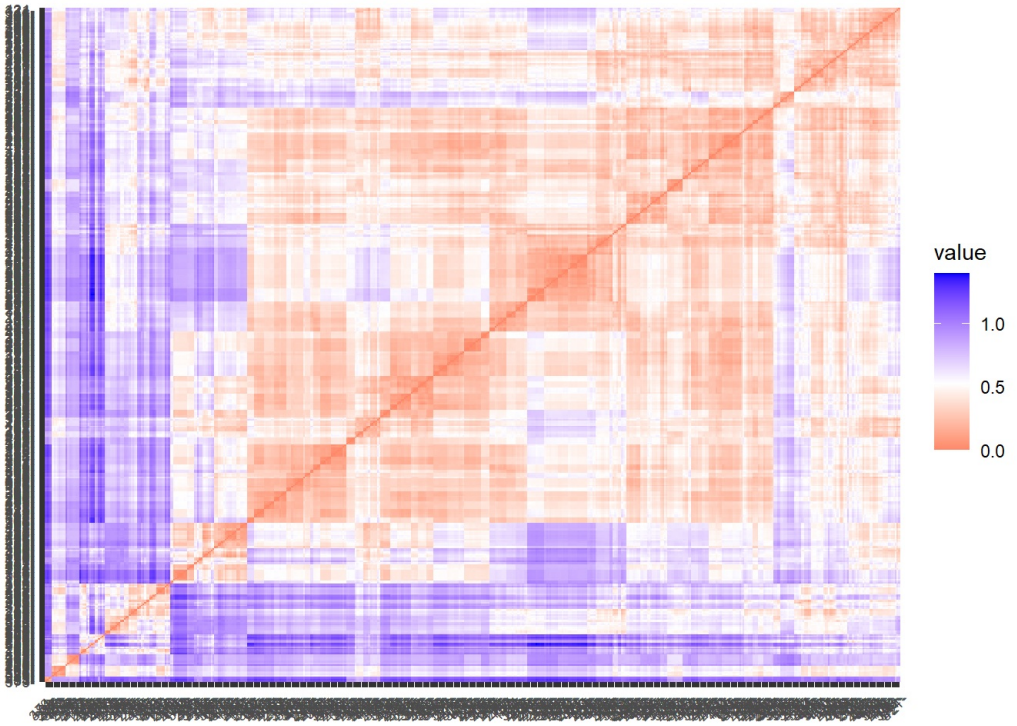
```
## Warning in rm(A, plot2, plot3, plot4, plot5, dist.cos, dist.eu, dist.man, :
## object 'dist.max' not found
```

```
## Warning in rm(A, plot2, plot3, plot4, plot5, dist.cos, dist.eu, dist.man, :
## object 'mds.cos' not found
```

```
## Warning in rm(A, plot2, plot3, plot4, plot5, dist.cos, dist.eu, dist.man, :
## object 'mds.max' not found
```

The Euclidean distance appears to give the most distinguishable clusters so we opted to use Euclidean distance moving forward.

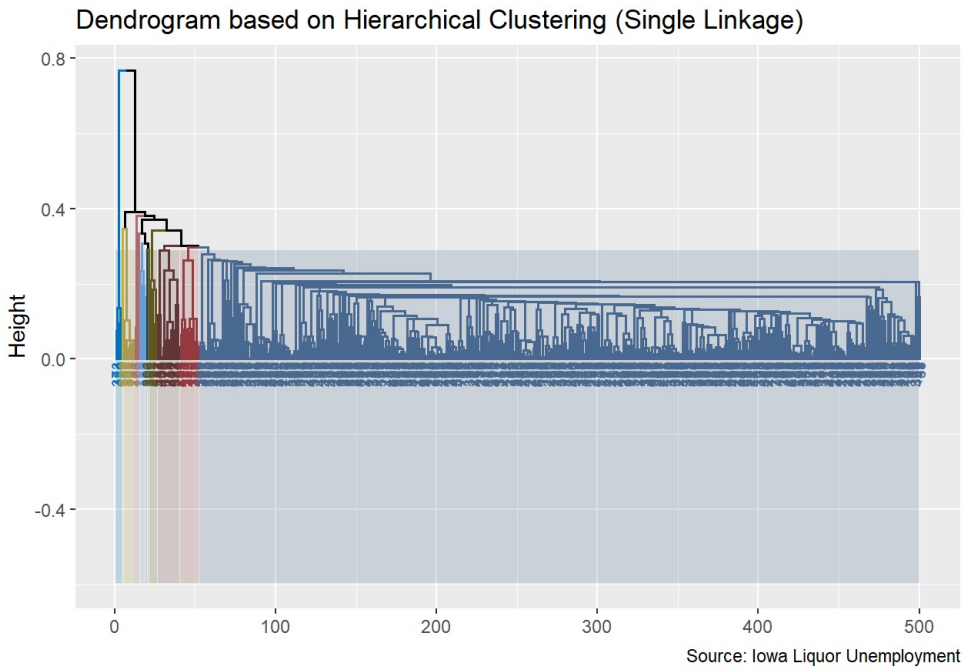
```
dd <- dist(sample_data_pca, method = "euclidean")
fviz_dist(dd, lab_size = 7)
```



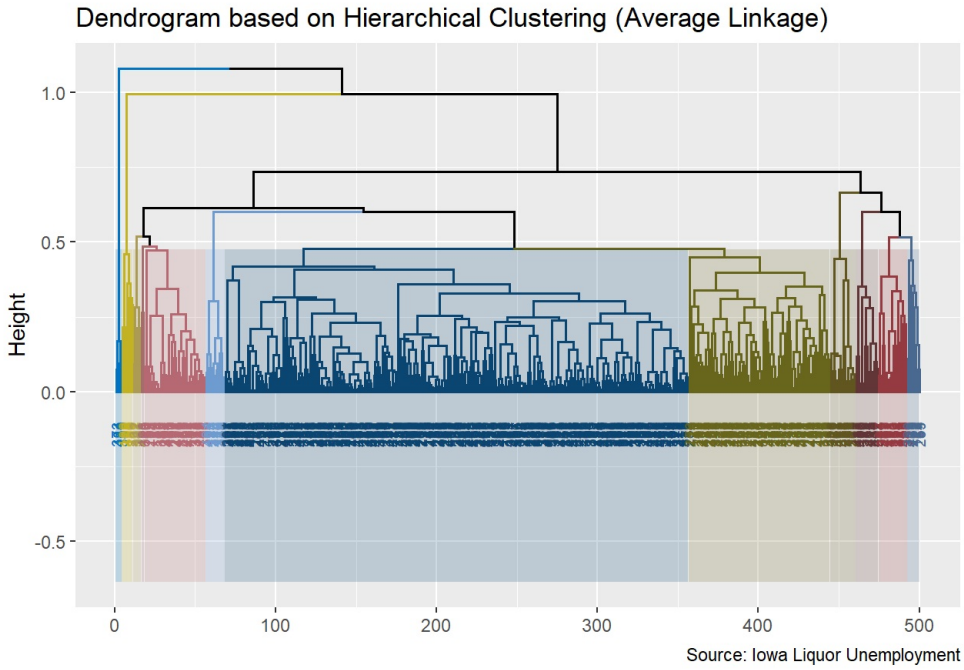
Hierarchical Clustering

To see the others also we cluster using 3 different methods: Single Linkage(MIN distance between clusters), Average Linkage(AVG distance between clusters) and Complete Linkage(MAX distance between clusters).

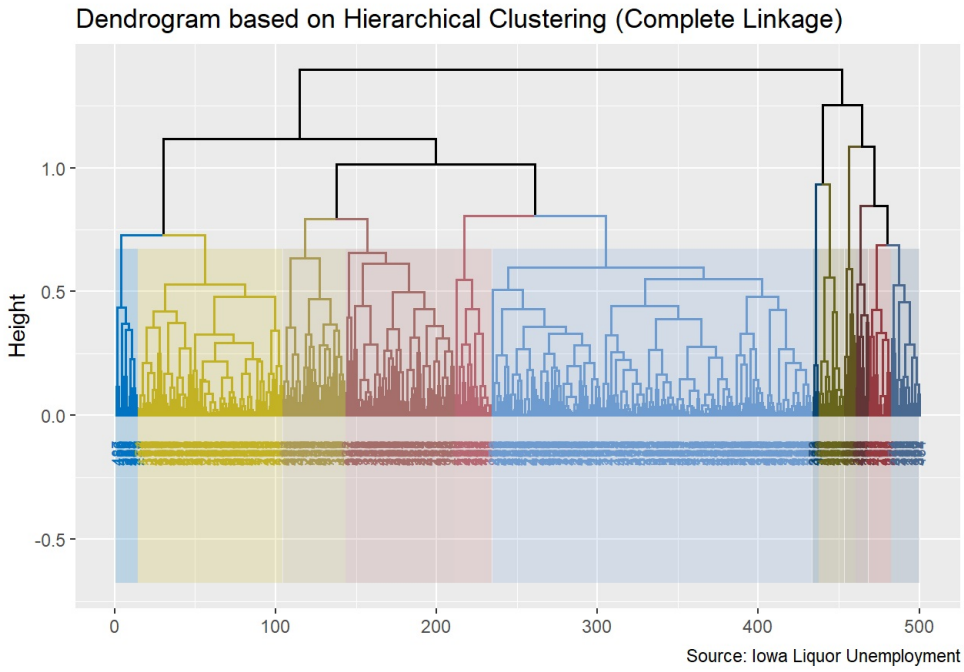
```
hcsingle <- hclust(dd, method = "single")
hcavg<- hclust(dd, method = "average")
hccomp <- hclust(dd, method = "complete")
rm(dd)
# Creating a draft of dendrogram by using fviz_dend() function:
  fviz_dend(hcsingle,
            k = 12,
            cex = 0.5,
            rect = TRUE,
            rect_fill = TRUE,
            horiz = FALSE,
            palette = "jco",
            rect_border = "jco",
            color_labels_by_k = TRUE) -> basic_plot
# Decorating the draft:
  basic_plot +
    theme_gray() +
    theme(plot.margin = unit(rep(0.7, 4), "cm")) +
    labs(title = "Dendrogram based on Hierarchical Clustering (Single Linkage)",
         caption = "Source: Iowa Liquor Unemployment")
```




```
  fviz_dend(hcavg,
            k = 12,
            cex = 0.5,
            rect = TRUE,
            rect_fill = TRUE,
            horiz = FALSE,
            palette = "jco",
            rect_border = "jco",
            color_labels_by_k = TRUE) -> basic_plot
# Decorating the draft:
  basic_plot +
    theme_gray() +
    theme(plot.margin = unit(rep(0.7, 4), "cm")) +
    labs(title = "Dendrogram based on Hierarchical Clustering (Average Linkage)",
         caption = "Source: Iowa Liquor Unemployment")
```



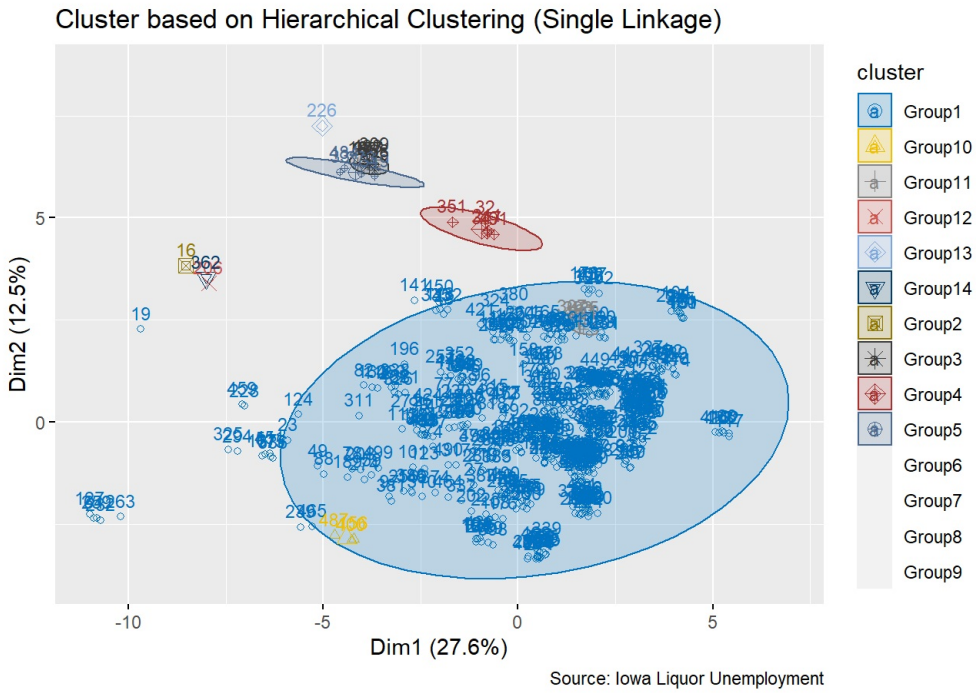
```
  fviz_dend(hccomp,
            k = 12,
            cex = 0.5,
            rect = TRUE,
            rect_fill = TRUE,
            horiz = FALSE,
            palette = "jco",
            rect_border = "jco",
            color_labels_by_k = TRUE) -> basic_plot
# Decorating the draft:
  basic_plot +
    theme_gray() +
    theme(plot.margin = unit(rep(0.7, 4), "cm")) +
    labs(title = "Dendrogram based on Hierarchical Clustering (Complete Linkage)",
         caption = "Source: Iowa Liquor Unemployment")
```




```
rm(basic_plot)
```

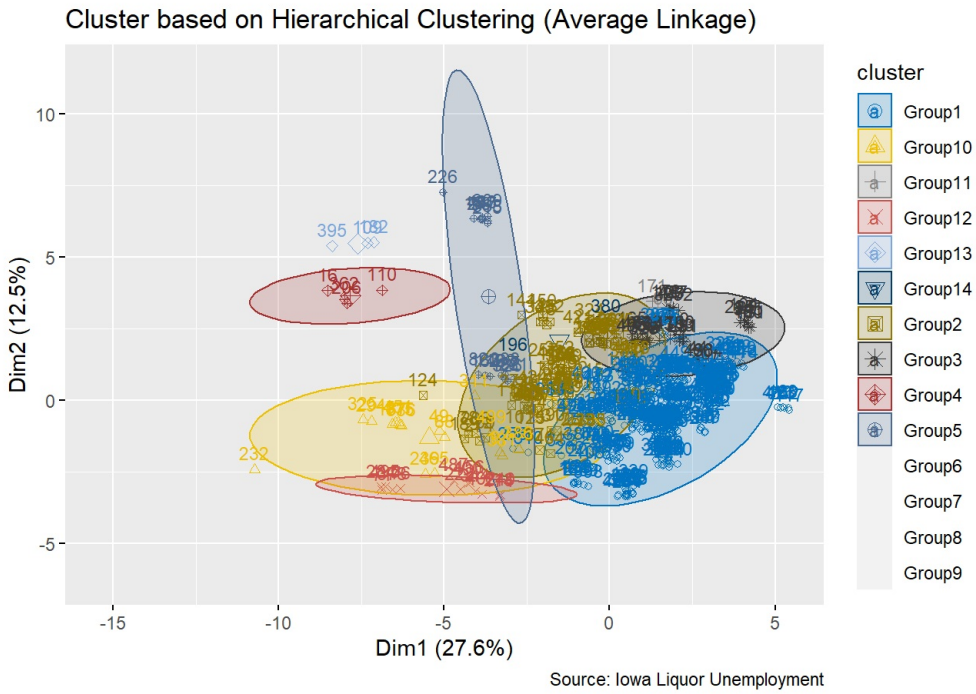
```
#Cut tree into 2 groups:
sub_grp <- cutree(hcsingle,14)
# Create plot of clusters:
fviz_cluster(list(data = sample_data, cluster = paste0("Group", sub_grp)),
             alpha = 1,
             palette = "jco",
             labelsize = 9,
             ellipse.type = "norm") -> cluster_plot
# Decorate the plot:
cluster_plot +
  theme(legend.position = "right") +
  theme(plot.margin = unit(rep(0.5, 4), "cm")) +
  labs(title = "Cluster based on Hierarchical Clustering (Single Linkage)",
       caption = "Source: Iowa Liquor Unemployment")
```

```
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
```

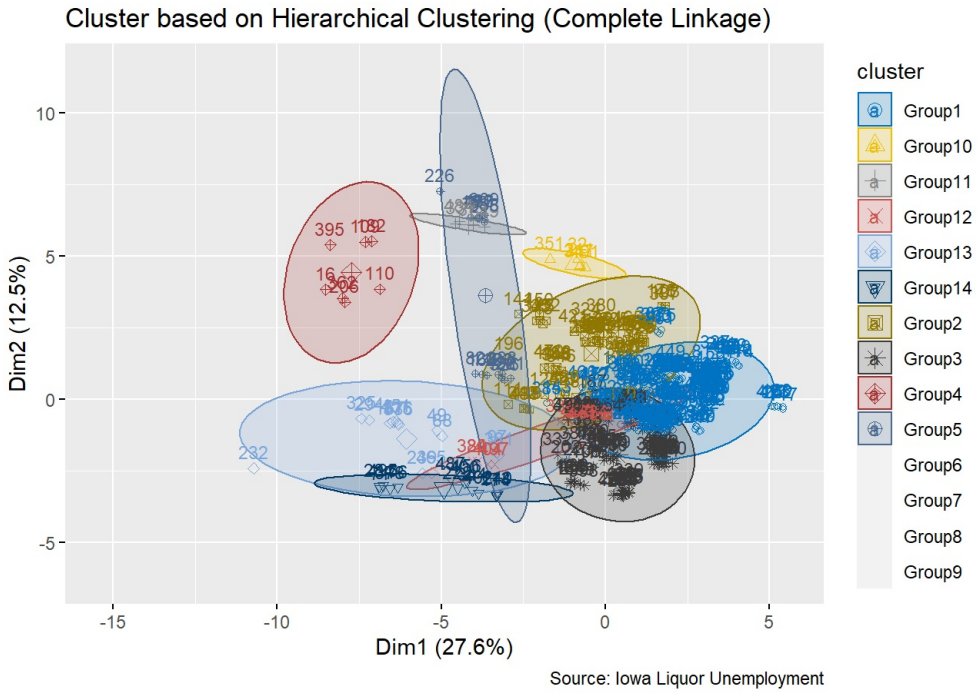



```
sub_grp <- cutree(hcavg, 14)
# Create plot of clusters:
fviz_cluster(list(data = sample_data, cluster = paste0("Group", sub_grp)),
             alpha = 1,
             palette = "jco",
             labelsize = 9,
             ellipse.type = "norm") -> cluster_plot
# Decorate the plot:
cluster_plot +
  theme(legend.position = "right") +
  theme(plot.margin = unit(rep(0.5, 4), "cm")) +
  labs(title = "Cluster based on Hierarchical Clustering (Average Linkage)",
       caption = "Source: Iowa Liquor Unemployment")
```

```
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
## Too few points to calculate an ellipse
```



```
sub_grp <- cutree(hccomp, 14)
# Create plot of clusters:
fviz_cluster(list(data = sample_data, cluster = paste0("Group", sub_grp)),
             alpha = 1,
             palette = "jco",
             labelsize = 9,
             ellipse.type = "norm") -> cluster_plot
# Decorate the plot:
cluster_plot +
  theme(legend.position = "right") +
  theme(plot.margin = unit(rep(0.5, 4), "cm")) +
  labs(title = "Cluster based on Hierarchical Clustering (Complete Linkage)",
       caption = "Source: Iowa Liquor Unemployment")
```

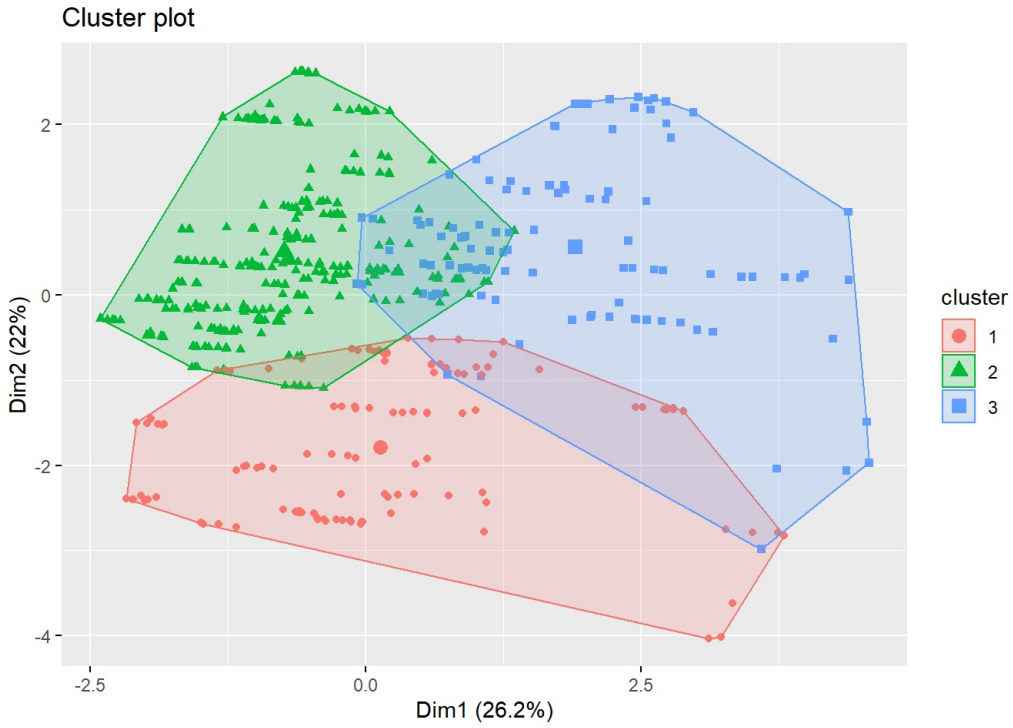



```
rm(cluster_plot,hcavg,hccomp,hcsingle,sub_grp)
```

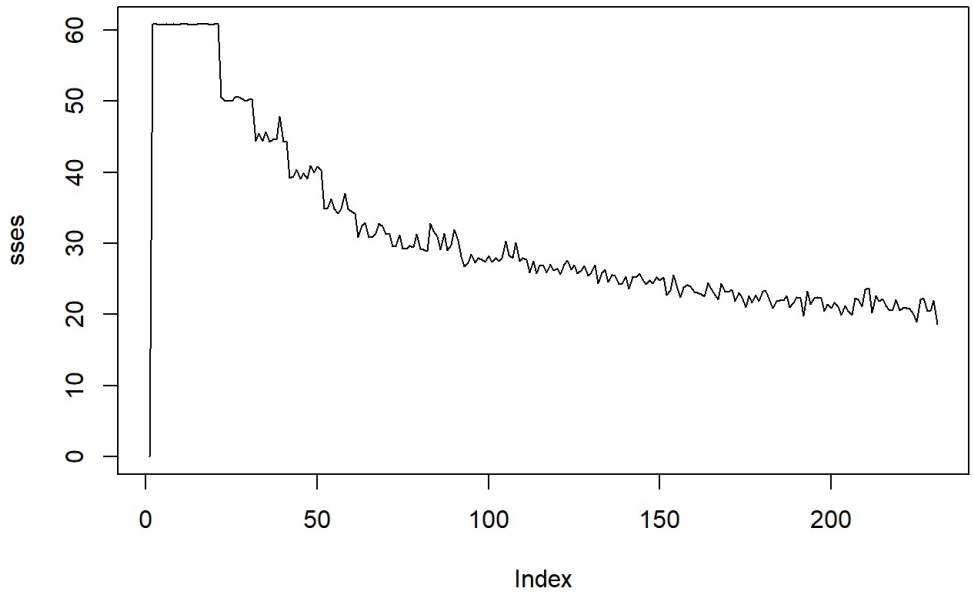
K-Means Clustering

We tried multiple instances of multiple k values to run the k-means algorithm. The best result is printed. The dunn indices for all runs can be seen.

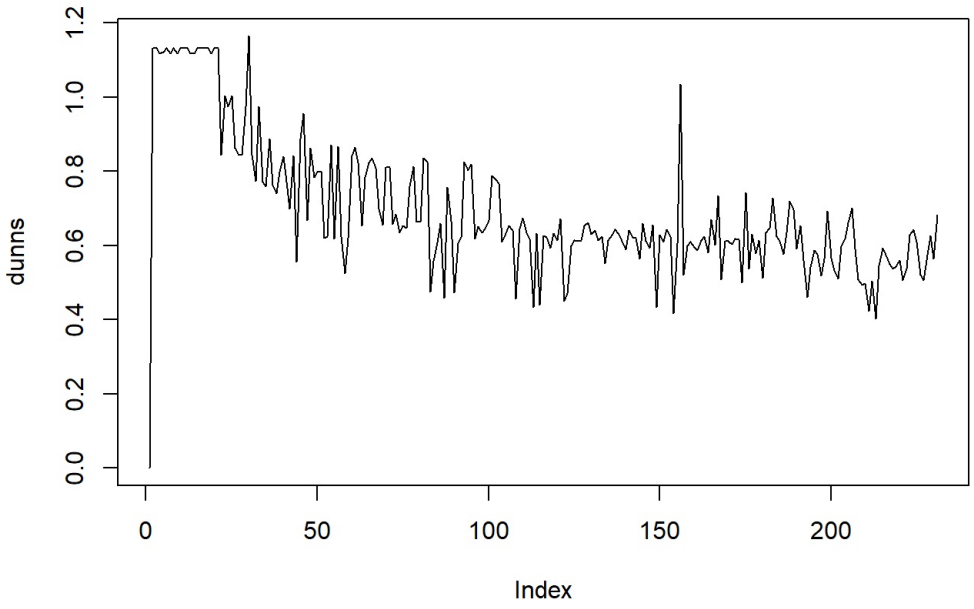
```
rm(.Random.seed, envir=globalenv())
is <- c(rep(2,20),rep(3,10),rep(4,10),rep(5,10),rep(6,10),rep(7,10),rep(8,20),rep(9,20),rep(10,20),rep(11,20),rep
(12,20),rep(13,20),rep(14,20),rep(15,20))
sses <- c(0)
dunns <- c(0)
maxdunn <- 0
for (i in is){
    kmeans <- kmeans(sample_data_pca,i)
    cluster <- fviz_cluster( kmeans, data = sample_data_pca, geom = "point")
    sses <- append(sses, kmeans$tot.withinss)
    if (i > 1){
        scattdata <- cls.scatt.data(sample_data_pca,kmeans$cluster)
        dunn <- clv.Dunn(scattdata, "average", "average")
        if(dunn > maxdunn){
          bestcluster <- cluster
          maxdunn <- dunn
        }
        dunns <- append(dunns,dunn)
    }
    else{
      bestcluster <- cluster
      maxdunn <- dunn
    }
}
print(bestcluster)
```




```
plot(sses,type="l")
```




```
# Judging from the dunn index comparison the best clustering is for k=5.
plot(dunns,type="l")
```



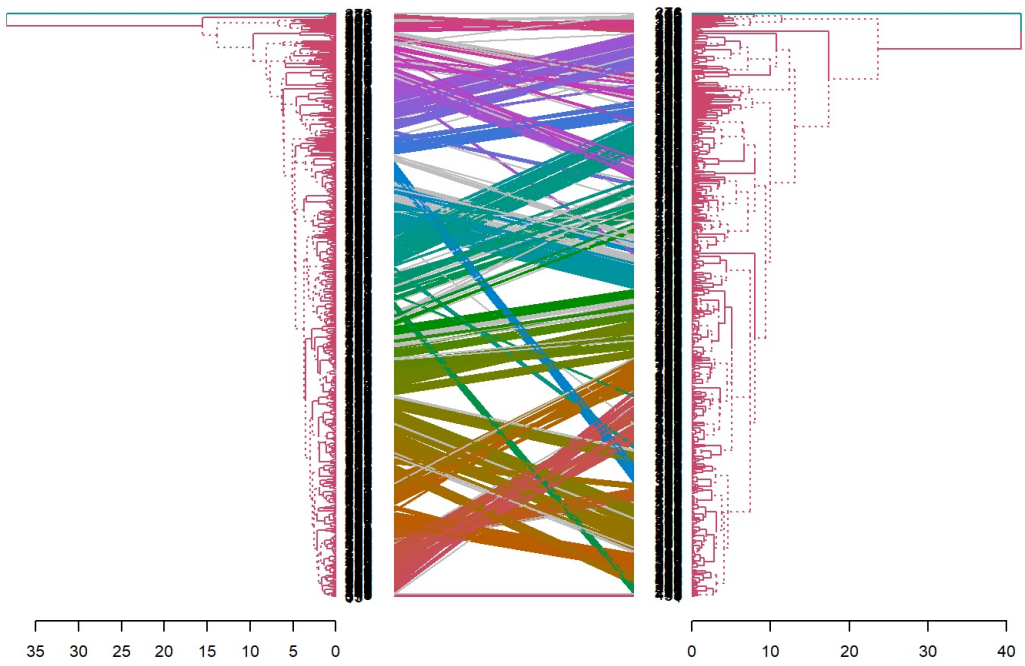

```
rm(i,is,sses,dunns,dunn,maxdunn,cluster,bestcluster,kmeans,scattdata)
```

Validation

AGNES and DIANA are Agglomerative and Divisive hierarchical clustering methods. As they have compatible working principles with each other, perfect clustering should be like straight lines.

```
agnes.cl <- agnes(sample_data_pca, stand = TRUE )
diana.cl <- diana(sample_data_pca, stand = TRUE )
clusters.ag <- cutree(agnes.cl , k = 2 )
clusters.di <- cutree(diana.cl , k = 2 )
```

```
dend1 <- as.dendrogram(agnes(sample_data_pca, stand = TRUE ))
dend2 <- as.dendrogram(diana(sample_data_pca, stand = TRUE ))
tanglegram(dend1,dend2, highlight_distinct_edges =TRUE, highlight_branches_lwd = FALSE, lwd = 1, k_branches = 2 )
```



In our case, we have seen a lot of straight and almost straight lines. There are some clusters which belong to different sub-branches; but after we cut the hierarchial tree, we can see they are still similar in upper-branches. It is important to mention that there are also some outliers that match which means there are points which are assigned to farther clusters.

```
clmethods <- c('hierarchical','kmeans')
rownames(sample_data) = seq(1, nrow(sample_data_pca) , by=1)
internal <- clValid(as.data.frame(sample_data_pca), nClust = 2:30, clMethods= clmethods, validation = 'internal'
)
summary(internal)
```

```
##
## Clustering Methods:
##  hierarchical kmeans
##
## Cluster sizes:
##  2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
##
## Validation Measures:
##                                   2        3        4        5        6        7        8        9       10
11       12       13       14       15       16       17       18       19       20       21       22       23
24       25       26       27       28       29       30
##
## hierarchical Connectivity    4.3825   9.9468  24.3651  26.7929  34.0603  35.8948  36.6738  37.7250  43.3405  4
7.8361  79.4218  83.2798  88.1556  89.3389  92.0595  93.7040  96.3869 103.9917 107.5135 111.4171 114.9782 114.978
2 117.6448 120.5976 123.6726 126.0468 130.8349 133.6972 139.0770
##              Dunn            0.5499   0.2795   0.1198   0.1212   0.1371   0.1403   0.1403   0.1403   0.1403
0.1403   0.1114   0.1114   0.1114   0.1114   0.1114   0.1114   0.1114   0.1121   0.1320   0.1320   0.1320   0.132
0   0.1320   0.1320   0.1320   0.1320   0.1320   0.1320   0.1320
##              Silhouette      0.5173   0.4702   0.3671   0.3402   0.2946   0.2677   0.2710   0.2635   0.2625
0.2423   0.2620   0.2605   0.2613   0.2521   0.2551   0.2560   0.2275   0.2811   0.2904   0.2908   0.2910   0.275
1   0.2738   0.2402   0.2389   0.2395   0.2401   0.2438   0.2475
## kmeans       Connectivity   82.3333  89.4885  99.5266  95.7409 112.6560 119.7226  99.3456 116.7544 130.3103 14
9.2694 130.5167 134.3746 148.3246 131.4496 134.1702 168.3401 177.2044 193.0655 192.9802 197.2377 200.7988 200.734
9 201.9810 176.0679 186.1476 173.4306 178.2187 181.5675 239.1000
##              Dunn            0.0288   0.0293   0.0314   0.0359   0.0582   0.0249   0.0526   0.0134   0.0277
0.0403   0.0500   0.0500   0.0510   0.0532   0.0532   0.0708   0.0720   0.0776   0.0494   0.0494   0.0494   0.042
1   0.0557   0.0509   0.0509   0.0591   0.0591   0.0591   0.0438
##              Silhouette      0.2885   0.2992   0.2804   0.2899   0.2700   0.2677   0.2697   0.2646   0.2629
0.2669   0.2829   0.2847   0.2838   0.2916   0.2955   0.2877   0.2968   0.2704   0.2829   0.2834   0.2835   0.290
1   0.2898   0.3053   0.3024   0.3124   0.3131   0.3085   0.3052
##
## Optimal Scores:
##
##              Score  Method       Clusters
## Connectivity 4.3825 hierarchical 2
## Dunn         0.5499 hierarchical 2
## Silhouette   0.5173 hierarchical 2
```

```
stab <- clValid(as.data.frame(sample_data_pca), nClust = 2:30, clMethods= clmethods, validation = 'stability')
summary(stab)
```

```
##
## Clustering Methods:
##  hierarchical kmeans
##
## Cluster sizes:
##  2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30
##
## Validation Measures:
##                          2      3      4      5      6      7      8      9     10     11     12     13     14
15     16     17     18     19     20     21     22     23     24     25     26     27     28     29     30
##
## hierarchical APN  0.0073 0.0474 0.0503 0.0656 0.0644 0.0904 0.0700 0.0755 0.0739 0.1430 0.1245 0.1244 0.1668 0
.2173 0.2383 0.2876 0.2914 0.1945 0.1852 0.1920 0.1969 0.1939 0.2279 0.2600 0.2633 0.2574 0.2619 0.2737 0.2935
##              AD   0.5152 0.5028 0.4661 0.4525 0.4267 0.4140 0.3996 0.3970 0.3862 0.3805 0.3615 0.3588 0.3534 0
.3467 0.3409 0.3388 0.3324 0.3092 0.3029 0.2985 0.2961 0.2925 0.2876 0.2834 0.2818 0.2768 0.2749 0.2699 0.2664
##              ADM  0.0064 0.0294 0.0512 0.0518 0.0725 0.0716 0.0643 0.0681 0.0552 0.0695 0.0929 0.0925 0.0932 0
.0930 0.0961 0.1068 0.1120 0.0991 0.0999 0.0974 0.0975 0.0965 0.0969 0.0940 0.0934 0.0904 0.0914 0.0926 0.0982
##              FOM  0.1327 0.1306 0.1272 0.1253 0.1225 0.1220 0.1212 0.1204 0.1174 0.1142 0.1141 0.1139 0.1137 0
.1127 0.1119 0.1115 0.1115 0.1102 0.1098 0.1090 0.1072 0.1067 0.1027 0.1007 0.1007 0.1000 0.0998 0.0991 0.0987
## kmeans       APN  0.0350 0.1006 0.1460 0.2557 0.2627 0.2450 0.2183 0.2480 0.2288 0.2865 0.2487 0.2215 0.2309 0
.3094 0.3280 0.3300 0.3007 0.3380 0.3248 0.3229 0.3794 0.3182 0.3703 0.3469 0.3212 0.3116 0.3161 0.3284 0.3396
##              AD   0.4621 0.4506 0.4210 0.4159 0.3955 0.3773 0.3578 0.3425 0.3341 0.3327 0.3222 0.3087 0.3055 0
.3055 0.2982 0.2919 0.2843 0.2821 0.2754 0.2729 0.2735 0.2620 0.2636 0.2558 0.2513 0.2488 0.2471 0.2462 0.2432
##              ADM  0.0407 0.0622 0.0867 0.1225 0.1138 0.1414 0.1223 0.1178 0.1118 0.1270 0.1143 0.0916 0.0919 0
.1108 0.1064 0.1092 0.1015 0.1126 0.1137 0.1141 0.1256 0.1079 0.1196 0.1161 0.1075 0.1117 0.1126 0.1160 0.1095
##              FOM  0.1291 0.1242 0.1253 0.1227 0.1220 0.1189 0.1155 0.1151 0.1118 0.1095 0.1090 0.1064 0.1057 0
.1076 0.1063 0.1042 0.1041 0.1035 0.1021 0.1025 0.1002 0.1000 0.0989 0.0969 0.0974 0.0973 0.0972 0.0974 0.0955
##
## Optimal Scores:
##
##     Score  Method       Clusters
## APN 0.0073 hierarchical 2
## AD  0.2432 kmeans       30
## ADM 0.0064 hierarchical 2
## FOM 0.0955 kmeans       30
```

Average Proportion of Non-overlap (APN) : proportion of observations not placed in the same cluster

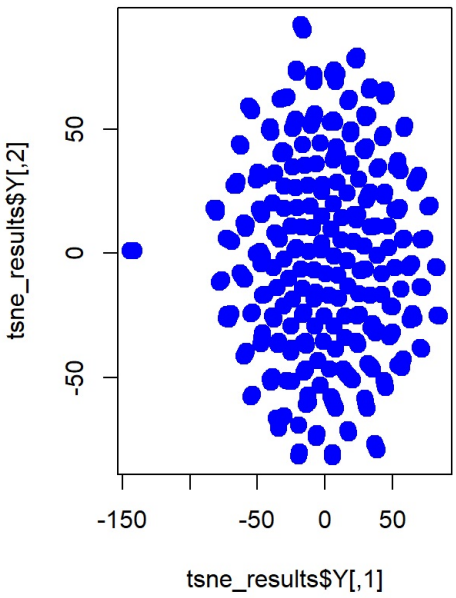
Average Distance (AD) : average distance between observations placed in the same cluster

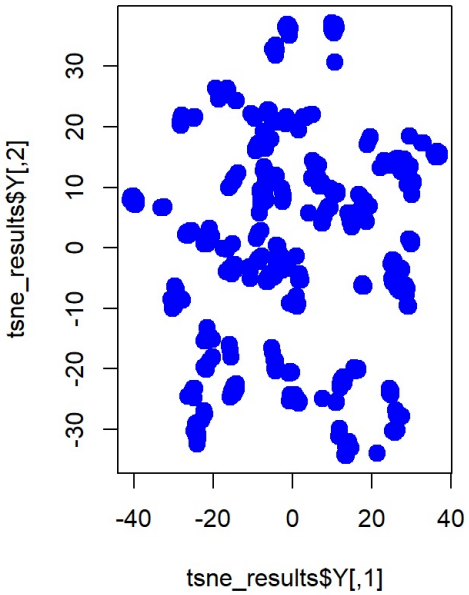
Average Distance between Means (ADM) : average distance between cluster centers

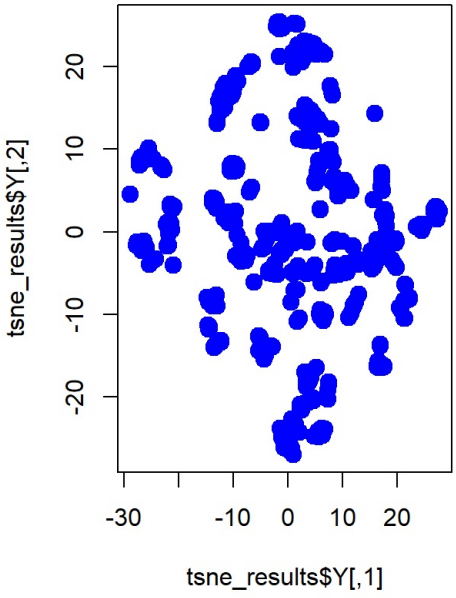
Figure of Merit (FOM) : average intra-cluster variance

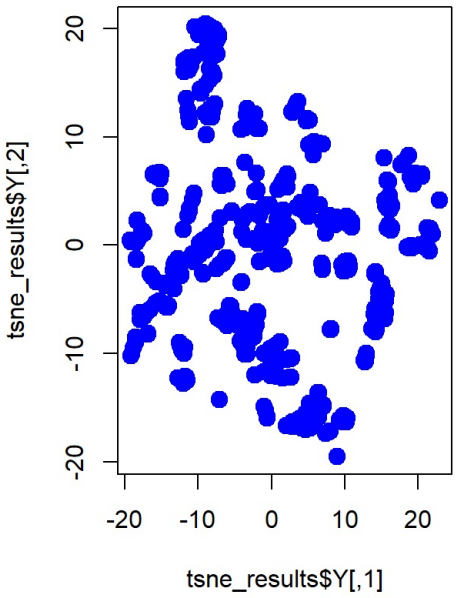
tSNE

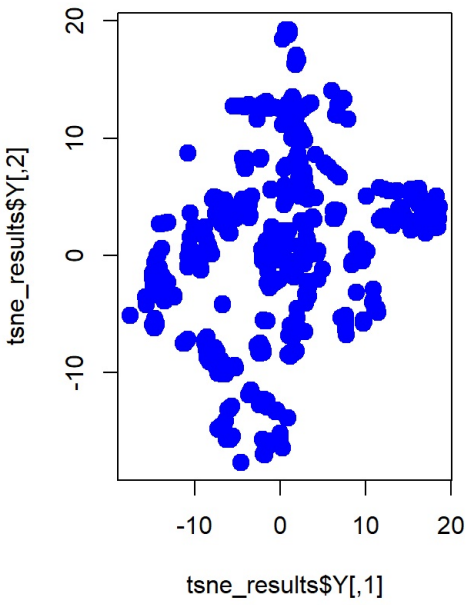
```
## Learning t-SNE Plotting
## Split IR into two objects:
## 1) Containing Measurements
## 2) Containing Types Of Houses
measurements_dataset <- sample_data_pca
#types_dataset <- Numeric_Dataset[ ,c("county_num")]

## Load the t-SNE library
library(Rtsne)

## Run the t-SNE algorithm and store the results into an object called tsne_results
for( i in seq.int(1,101,10)){
tsne_results <- Rtsne(measurements_dataset, perplexity=i, check_duplicates = FALSE) #Value of perplexity can be c
hanged
## Generate the t_SNE plot
par(mfrow=c(1,2)) # To plot two images side-by-side
plot(tsne_results$Y, col = "blue", pch = 19, cex = 1.5) # Plotting the first image
#plot(tsne_results$Y, col = "black", bg= types_dataset, pch = 21, cex = 1.5) # Second plot: Color the plot by the
types of houses (bg= types_dataset)
}
```

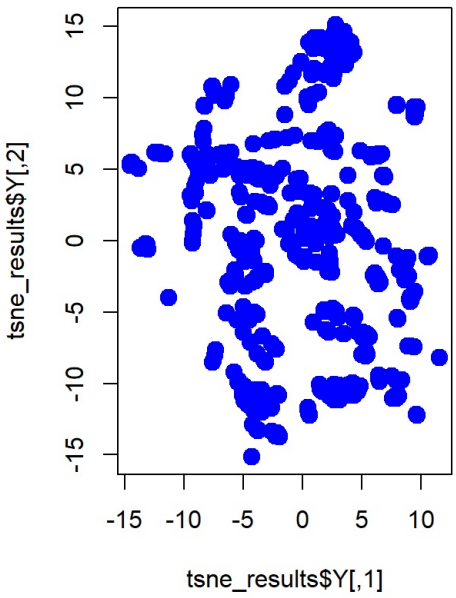

tsne_results$Y[,2]
50
0
-50
-150
-50
0
50
tsne_results$Y[,1]

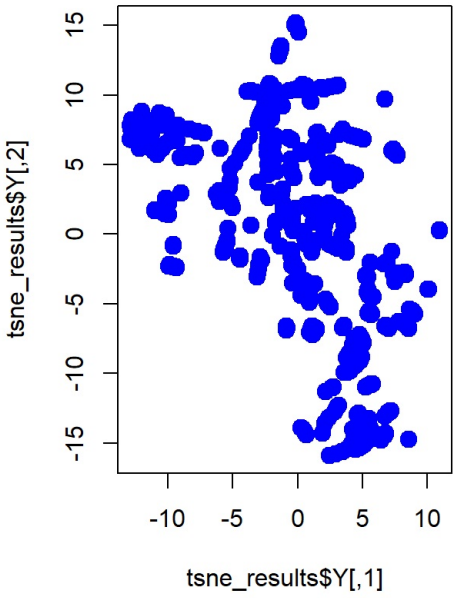

tsne_results$Y[,2]
30
20
10
0
-10
-20
-30
-40
-20
0
20
40
tsne_results$Y[,1]

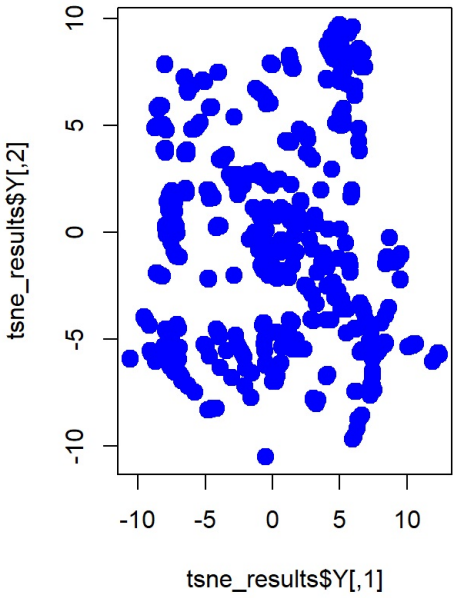

tsne_results$Y[,2]
20
10
0
-10
-20
-30
-10
0
10
20
tsne_results$Y[,1]

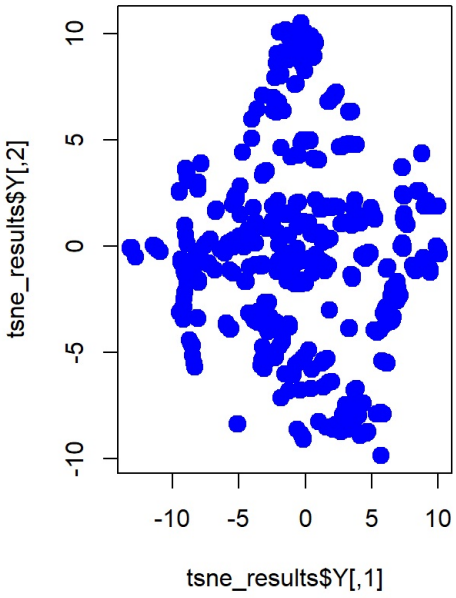

tsne_results$Y[,2]
20
10
0
-10
-20
-20
-10
0
10
20
tsne_results$Y[,1]

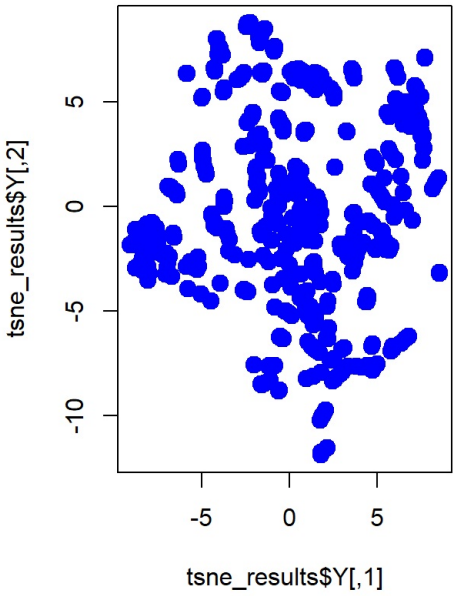

tsne_results$Y[,2]
20
10
0
-10
-10
0
10
20
tsne_results$Y[,1]


tsne_results$Y[,2]
15
10
5
0
-5
-10
-15
-15
-10
-5
0
5
10
tsne_results$Y[,1]


tsne_results$Y[,2]
15
10
5
0
-5
-10
-15
-10
-5
0
5
10
tsne_results$Y[,1]

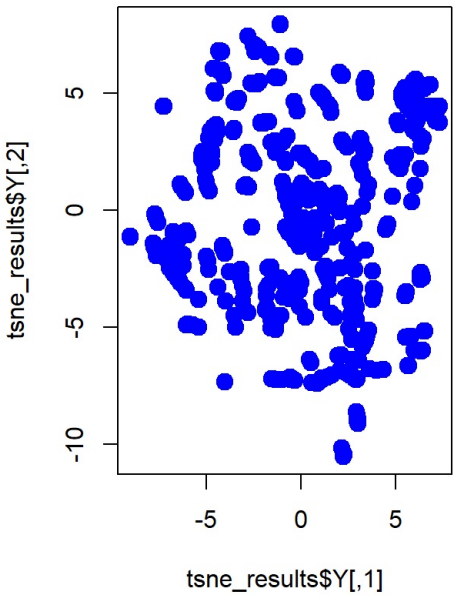

tsne_results$Y[,2]
10
5
0
-5
-10
-10
-5
0
5
10
tsne_results$Y[,1]


tsne_results$Y[,2]
10
5
0
-5
-10
-10
-5
0
5
10
tsne_results$Y[,1]

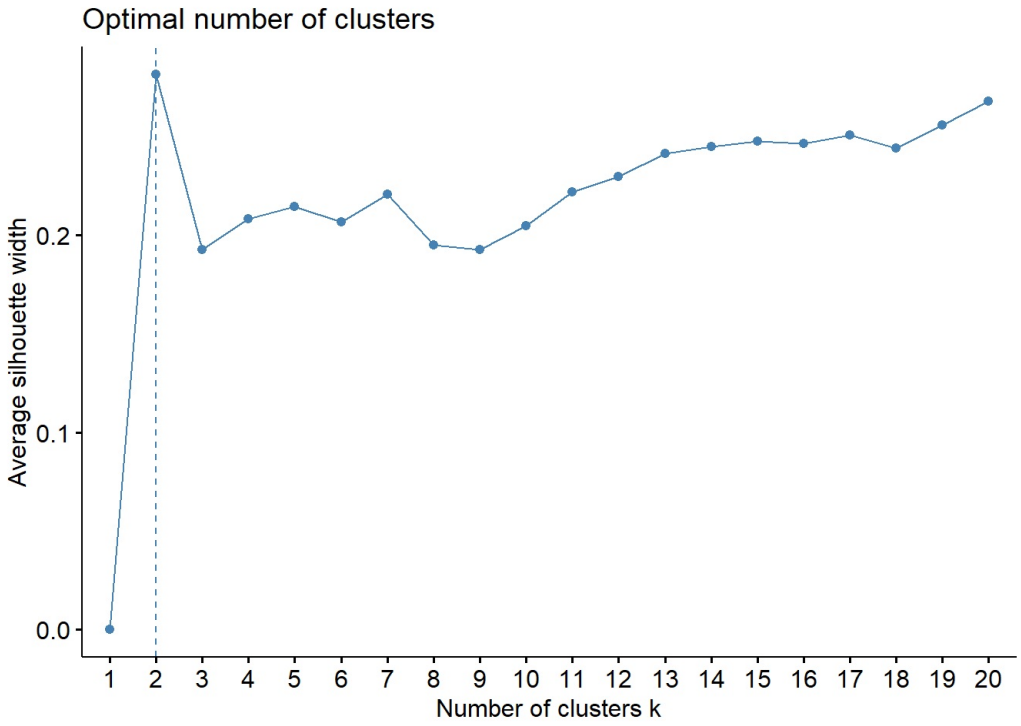



```
rm(tsne_results,types_dataset,measurements_dataset)
```

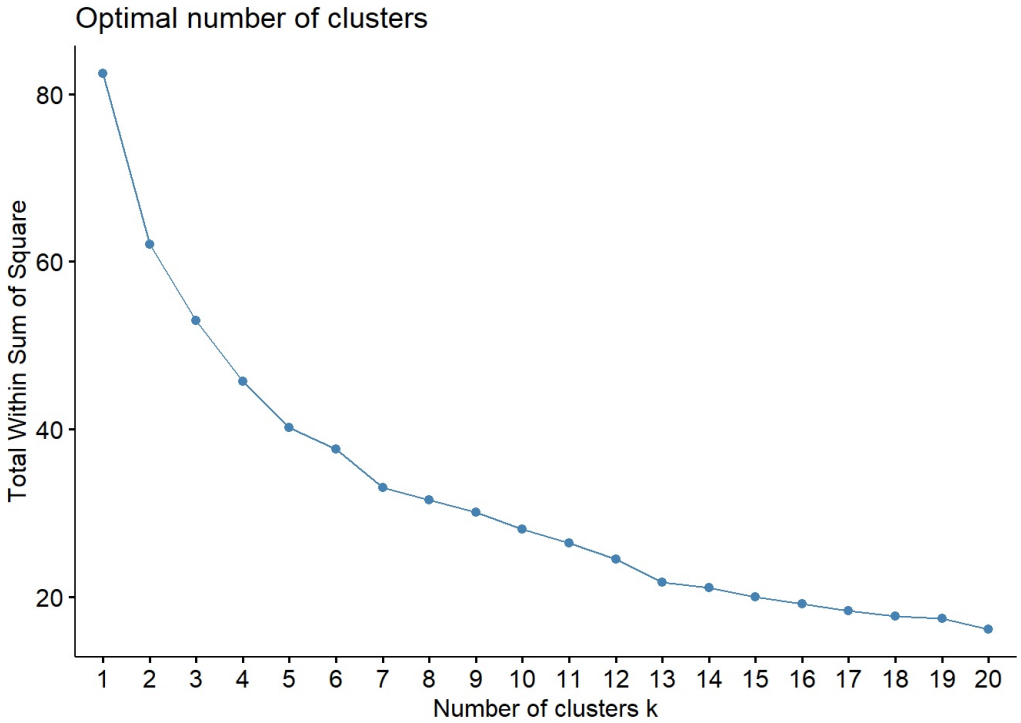
```
## Warning in rm(tsne_results, types_dataset, measurements_dataset): object
## 'types_dataset' not found
```




```
fviz_nbclust(sample_data_pca, FUNcluster = cluster::pam, method = c("silhouette"), k.max = 20, nboot = 100,)
```

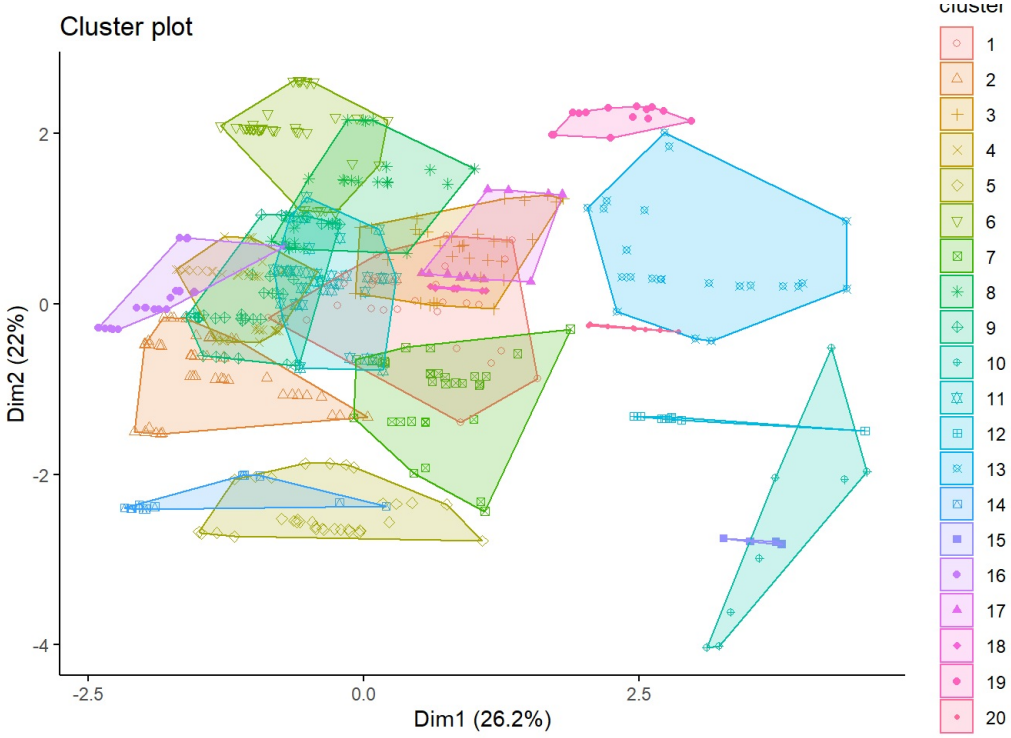



```
fviz_nbclust(sample_data_pca, FUNcluster = cluster::pam, method = c("wss"), k.max = 20, nboot = 100,)
```



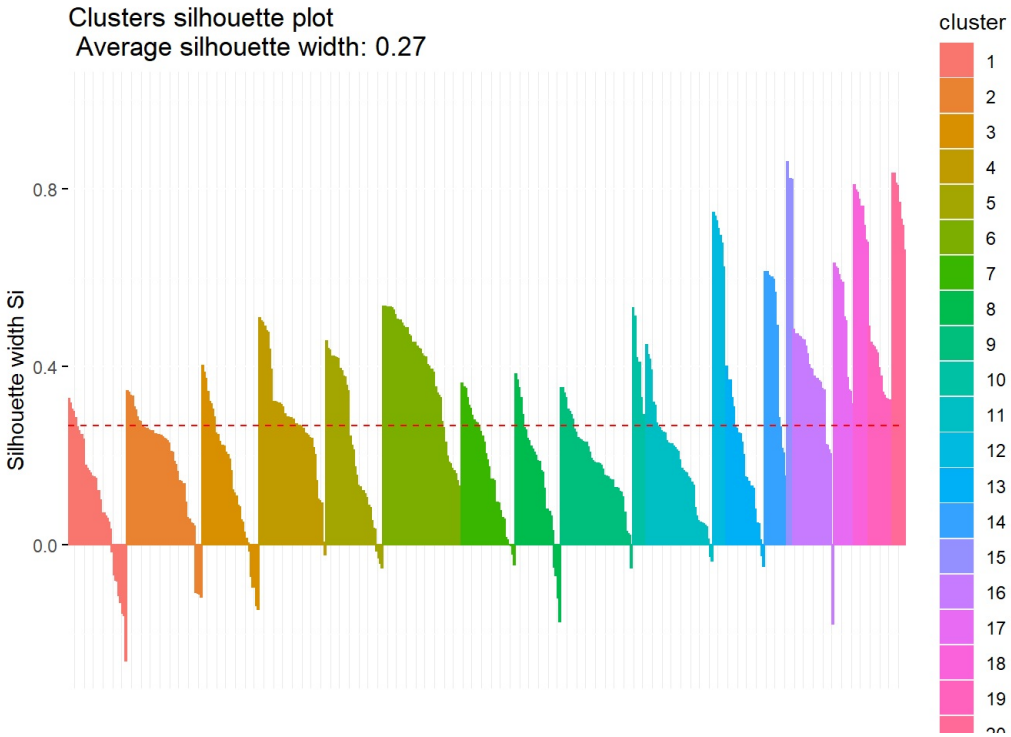
PAM K_medioids

```
library(cluster)
k_value <- 20
#make this example reproducible
#perform k-medoids clustering with k = k_value clusters
kmed <- pam(sample_data_pca, k_value, metric = "euclidean", stand = FALSE)
#plot results of final k-medoids model
fviz_cluster(kmed, ellipse.type = "convex", geom = "point", ggtheme=theme_classic())
```




```
fviz_silhouette(kmed)
```

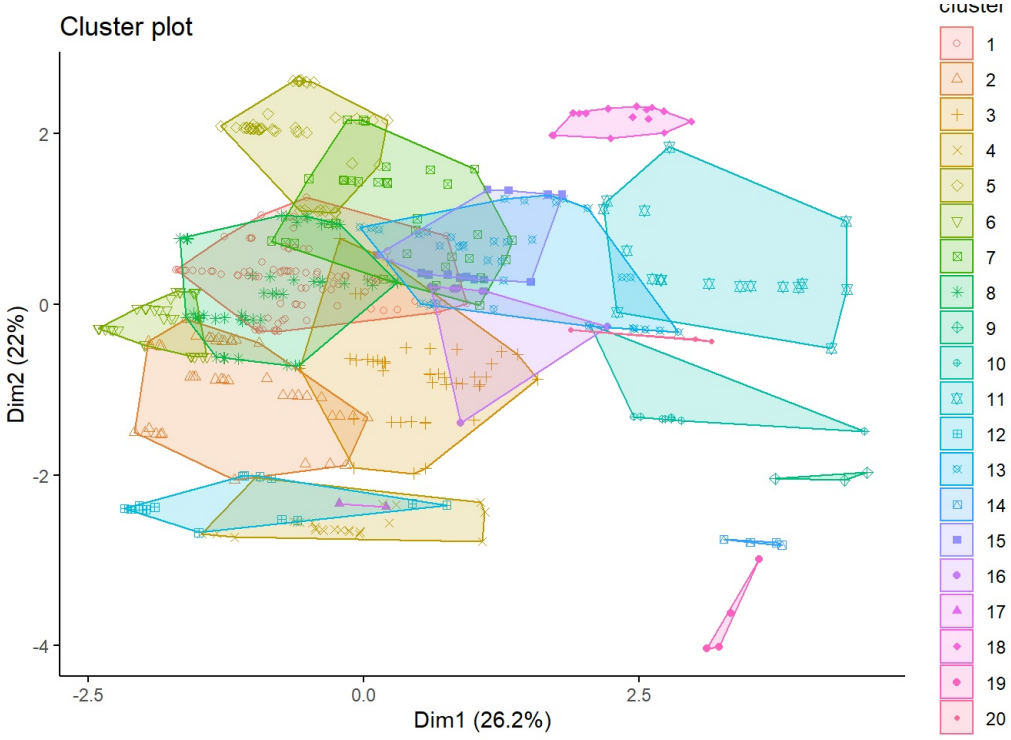
```
##    cluster size ave.sil.width
## 1        1   35          0.10
## 2        2   45          0.19
## 3        3   34          0.15
## 4        4   40          0.29
## 5        5   34          0.22
## 6        6   47          0.39
## 7        7   32          0.17
## 8        8   27          0.16
## 9        9   43          0.19
## 10      10    8          0.37
## 11      11   40          0.19
## 12      12    8          0.65
## 13      13   23          0.18
## 14      14   13          0.45
## 15      15    4          0.83
## 16      16   24          0.36
## 17      17   12          0.51
## 18      18    9          0.75
## 19      19   14          0.39
## 20      20    8          0.77
```




```
rm(k_value, kmed)
```

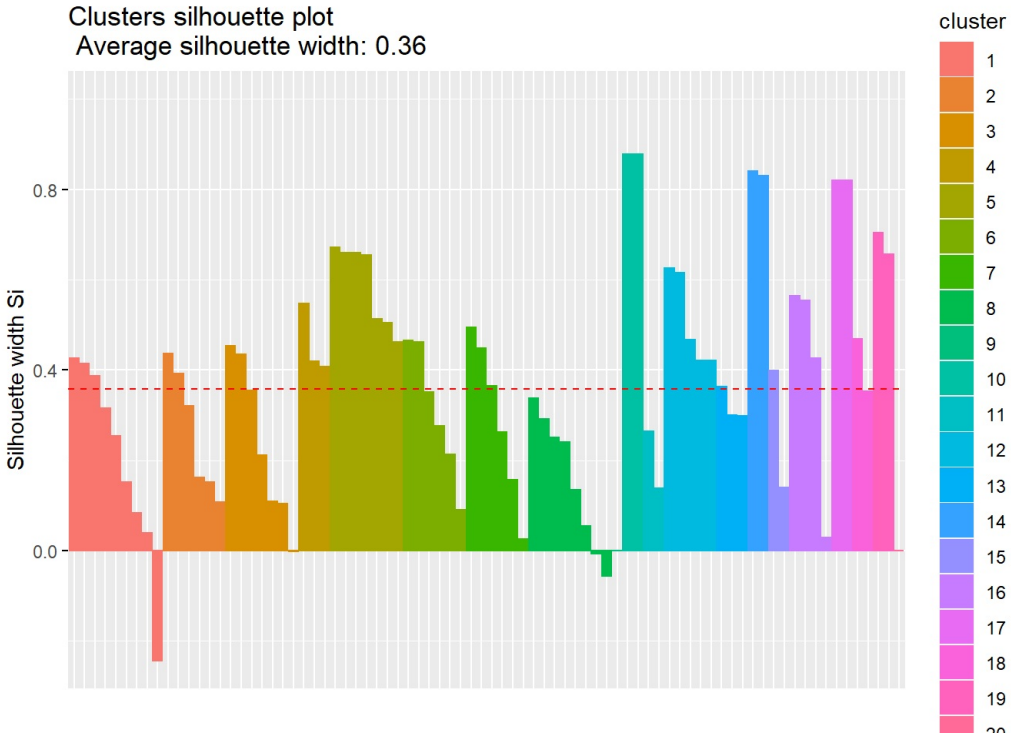
Clara K_medioids

```
k_value <- 20
#make this example reproducible
#perform k-medoids clustering with k = k_value clusters
kmed <- clara(sample_data_pca, k_value, metric = "euclidean", samples = 10, pamLike = TRUE)
#plot results of final k-medoids model
fviz_cluster(kmed, ellipse.type = "convex", geom = "point", ggtheme=theme_classic())
```




```
fviz_silhouette(kmed)
```

```
##    cluster size ave.sil.width
## 1        1    9          0.20
## 2        2    6          0.26
## 3        3    7          0.24
## 4        4    3          0.46
## 5        5    7          0.59
## 6        6    6          0.31
## 7        7    6          0.29
## 8        8    8          0.16
## 9        9    1          0.00
## 10      10    2          0.88
## 11      11    2          0.20
## 12      12    5          0.51
## 13      13    3          0.32
## 14      14    2          0.84
## 15      15    2          0.27
## 16      16    4          0.39
## 17      17    2          0.82
## 18      18    2          0.41
## 19      19    2          0.68
## 20      20    1          0.00
```




```
rm(k_value, kmed)
```

References

tSNE: https://ajitjohnson.com/tsne-for-biologist-tutorial/ (https://ajitjohnson.com/tsne-for-biologist-tutorial/)