```
library(tidyverse)

## -- Attaching packages --------------------------------------- tidyverse
1.3.1 --

## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.4     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   2.0.1     v forcats 0.5.1

## -- Conflicts ------------------------------------------
tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()

library(tidyr)
library(ggplot2)
library(gridExtra)

##
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':
##
##     combine

library(GGally)

## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2

library(coefplot)
library(corrplot)

## corrplot 0.90 loaded

library(tmap)

## Warning: package 'tmap' was built under R version 4.1.2

library(tmaptools)

## Warning: package 'tmaptools' was built under R version 4.1.2

library(sf)

## Warning: package 'sf' was built under R version 4.1.2

## Linking to GEOS 3.9.1, GDAL 3.2.1, PROJ 7.2.1

library(leaflet)

## Warning: package 'leaflet' was built under R version 4.1.2
```

```
library(rnaturalearth)

## Warning: package 'rnaturalearth' was built under R version 4.1.2

library(rnaturalearthdata)

## Warning: package 'rnaturalearthdata' was built under R version 4.1.2

library(rgeos)

## Warning: package 'rgeos' was built under R version 4.1.2

## Loading required package: sp

## rgeos version: 0.5-8, (SVN revision 679)
##  GEOS runtime version: 3.9.1-CAPI-1.14.2
##  Please note that rgeos will be retired by the end of 2023,
## plan transition to sf functions using GEOS at your earliest convenience.
##  GEOS using OverlayNG
##  Linking to sp version: 1.4-6
##  Polygon checking: TRUE

library(maps)

## Warning: package 'maps' was built under R version 4.1.2

##
## Attaching package: 'maps'

## The following object is masked from 'package:purrr':
##
##     map

library(mapproj)

## Warning: package 'mapproj' was built under R version 4.1.2

library(mapdata)

## Warning: package 'mapdata' was built under R version 4.1.2

library(ggmap)

## Warning: package 'ggmap' was built under R version 4.1.2

## Google's Terms of Service: https://cloud.google.com/maps-platform/terms/.

## Please cite ggmap if you use it! See citation("ggmap") for details.

library(usmap)

## Warning: package 'usmap' was built under R version 4.1.2

library(mltools)
```

```
## Warning: package 'mltools' was built under R version 4.1.2

##
## Attaching package: 'mltools'

## The following object is masked from 'package:tidyr':
##
##     replace_na

library(data.table)

##
## Attaching package: 'data.table'

## The following objects are masked from 'package:dplyr':
##
##     between, first, last

## The following object is masked from 'package:purrr':
##
##     transpose

install.packages("fiftystater_1.0.1.tar.gz",
                 repos = NULL, type = "source")

## Installing package into 'R/win-library/4.1'
## (as 'lib' is unspecified)

library(fiftystater)

#reading the data in:
d<-read_csv('project.csv')

## Rows: 72836 Columns: 19

## -- Column specification --------------------------------------------------
------
## Delimiter: ","
## chr  (8): STATE, ST_ABBR, County, Location, Total Score, Hardest Hit Area
(H...
## dbl (11): STCNTY, FIPS, County FIPS, Max Possible Score, HHA Score, Low
Inco...

##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this
message.

d[d$`Total Score`=="Manual Verification Required",]<- NA
sum(is.na(d$`Total Score`))

## [1] 963
```

```
d %>%
  mutate(across(c(`Total Score`),na_if, "Manual Verification Required"))
```

```
## # A tibble: 72,836 x 19
##    STATE      ST_ABBR STCNTY County    FIPS `County FIPS` Location `Total
Score`
##    <chr>      <chr>    <dbl> <chr>    <dbl>         <dbl> <chr>    <chr>
##  1 ALASKA     AK        2170 Matan~ 2.17e 9          2170 Census ~ 50
##  2 MICHIGAN   MI       26035 Clare  2.60e10         26035 Census ~ 50
##  3 MICHIGAN   MI       26073 Isabe~ 2.61e10         26073 Census ~ 50
##  4 MICHIGAN   MI       26073 Isabe~ 2.61e10         26073 Census ~ 50
##  5 MICHIGAN   MI       26073 Isabe~ 2.61e10         26073 Census ~ 50
##  6 MINNESOTA  MN       27005 Becker 2.70e10         27005 Census ~ 50
##  7 MINNESOTA  MN       27021 Cass   2.70e10         27021 Census ~ 50
##  8 MINNESOTA  MN       27095 Mille~ 2.71e10         27095 Census ~ 50
##  9 NEW MEXICO NM       35045 San J~ 3.50e10         35045 Census ~ 50
## 10 NEW MEXICO NM       35045 San J~ 3.50e10         35045 Census ~ 50
## # ... with 72,826 more rows, and 11 more variables: Max Possible Score
<dbl>,
## #   Hardest Hit Area (HHA) <chr>, HHA Score <dbl>,
## #   Low Income Area (LIA) County SAIPE - (Poverty Percentage) <dbl>,
## #   Low Income Area (LIA) County SAIPE- Score <dbl>,
## #   Low Income Area (LIA) Census Tract (Poverty Percentage) <dbl>,
## #   Low Income Area (LIA) Census Tract - Score <dbl>,
## #   Tribal Community (1 if yes) <chr>, ...
```

```
is.na(d) <- d == "Manual Verification Required"

table(d$`Low Income Area (LIA) County SAIPE - (Poverty Percentage)`)
```

```
## 
## 0.027 0.031 0.035 0.038  0.04 0.041 0.042 0.043 0.044 0.045 0.046 0.047
0.048
##    61    65     7    11    39    18    62    71    61    97   129    99
46
## 0.049  0.05 0.051 0.052 0.053 0.054 0.055 0.056 0.057 0.058 0.059  0.06
0.061
##    35    60   168    87   270   282   132   320   530   245   185   776
790
## 0.062 0.063 0.064 0.065 0.066 0.067 0.068 0.069  0.07 0.071 0.072 0.073
0.074
##   294   143   125   412   260   171   366   485   193   191   739   444
235
## 0.075 0.076 0.077 0.078 0.079  0.08 0.081 0.082 0.083 0.084 0.085 0.086
0.087
##   272    92   433   777   624   249   292   586   320   379   291   609
480
## 0.088 0.089  0.09 0.091 0.092 0.093 0.094 0.095 0.096 0.097 0.098 0.099
0.1
##   437   868   793   397   448   597   883   968   218   647   271   483
```

```
170
## 0.101 0.102 0.103 0.104 0.105 0.106 0.107 0.108 0.109  0.11 0.111 0.112
0.113
##   331   787  1028   388   537   313   676   986   264   908   226   686
1098
## 0.114 0.115 0.116 0.117 0.118 0.119  0.12 0.121 0.122 0.123 0.124 0.125
0.126
##   808   457   318   475   253   251   720   574  1274   649   475   328
1082
## 0.127 0.128 0.129  0.13 0.131 0.132 0.133 0.134 0.135 0.136 0.137 0.138
0.139
##   551   179   576  1832   327   146  1483  2697  1037   320   618   936
391
##  0.14 0.141 0.142 0.143 0.144 0.145 0.146 0.147 0.148 0.149  0.15 0.151
0.152
##  1240   621   492   355   247   208   821   123   253   498   911   269
954
## 0.153 0.154 0.155 0.156 0.157 0.158 0.159  0.16 0.161 0.162 0.163 0.164
0.165
##   373   273   343   151   646   226   247   421   278   815   171   169
201
## 0.166 0.167 0.168 0.169  0.17 0.171 0.172 0.173 0.174 0.175 0.176 0.177
0.178
##   300   367   133   515   163    79   376   201    69   199   140  1064
191
## 0.179  0.18 0.181 0.182 0.183 0.184 0.185 0.186 0.187 0.188 0.189  0.19
0.191
##   132   306    99    55   105   289   148   121   116   256   207   289
104
## 0.192 0.193 0.194 0.195 0.196 0.197 0.198 0.199   0.2 0.201 0.202 0.203
0.204
##   187    61    51    54   106    62   636   126    64   134    34    81
404
## 0.205 0.206 0.207 0.208 0.209  0.21 0.211 0.212 0.213 0.214 0.215 0.216
0.217
##   246   144    23   158   121    60    21    18    13    86    83    60
110
## 0.218 0.219  0.22 0.221 0.222 0.223 0.224 0.225 0.226 0.227 0.228 0.229
0.23
##    26    95    52    43    20    13    45    42    44    31    27    44
422
## 0.231 0.232 0.233 0.234 0.235 0.236 0.237 0.238 0.239  0.24 0.241 0.242
0.243
##    25    60    47    19   200    18     4   105    61    32    82    31
10
## 0.244 0.245 0.246 0.248 0.249 0.251 0.252 0.253 0.254 0.255 0.256 0.257
0.258
##    41     3     2     8    38    22    40     4    32    96     2    33
23
## 0.259  0.26 0.261 0.262 0.263 0.264 0.265 0.266 0.267 0.268 0.269 0.271
```

```
0.273
##     4    23     5   347    38     5    12    20    24     7   127    21
15
## 0.274 0.275 0.276 0.277 0.278 0.279  0.28 0.281 0.282 0.285 0.286 0.288
0.289
##    20     5    33    12    14     9     2     3     7     7     8     1
19
##  0.29 0.292 0.293 0.294 0.295 0.296 0.297 0.299   0.3 0.301 0.302 0.303
0.305
##     2     9    12    12    10    17     5     1     3    18     3     9
13
## 0.307  0.31 0.311 0.312 0.315 0.316 0.317  0.32 0.321 0.323 0.325 0.326
0.333
##     5    12    19     4    39     4     3     6     2     3    27     6
6
## 0.334 0.337 0.338 0.339  0.34 0.342 0.344 0.345 0.347 0.349  0.35 0.354
0.355
##    16    19    13     3     7     2     4     4     2     3     3     4
4
## 0.357 0.358 0.359 0.364 0.366 0.371 0.375 0.379 0.382 0.384 0.398   0.4
0.401
##     8     1     2    10     8     3     3     4     7     3     1     1
3
## 0.403 0.411 0.434
##     2     5     2
```

```
summary(d$`Total Score`)
```

```
##    Length     Class      Mode
##     72836 character character
```

```
sapply(d, class)
```

```
##                                                      STATE
##                                                "character"
##                                                    ST_ABBR
##                                                "character"
##                                                     STCNTY
##                                                  "numeric"
##                                                     County
##                                                "character"
##                                                       FIPS
##                                                  "numeric"
##                                                County FIPS
##                                                  "numeric"
##                                                   Location
##                                                "character"
##                                                Total Score
##                                                "character"
##                                         Max Possible Score
##                                                  "numeric"
##                                     Hardest Hit Area (HHA)
##                                                "character"
##                                                  HHA Score
##                                                  "numeric"
## Low Income Area (LIA) County SAIPE - (Poverty Percentage)
##                                                  "numeric"
##                  Low Income Area (LIA) County SAIPE- Score
##                                                  "numeric"
##   Low Income Area (LIA) Census Tract (Poverty Percentage)
##                                                  "numeric"
##                 Low Income Area (LIA) Census Tract - Score
##                                                  "numeric"
##                                Tribal Community (1 if yes)
##                                                "character"
##                   Tribal Community Score (Geographic Only)
##                                                "character"
##                                                      Rural
##                                                  "numeric"
##                                              Rural - Score
##                                                  "numeric"
```

```
sum(is.na(d$`Low Income Area (LIA) County SAIPE - (Poverty Percentage)`))
```

```
## [1] 963
```

```
d<-na.omit(d)

#Conversion of classes:
d$`Hardest Hit Area (HHA)`=as.factor(d$`Hardest Hit Area (HHA)`)
u<-unique(d[c("STATE", "ST_ABBR")])
names(d)
```

```
##  [1] "STATE"
##  [2] "ST_ABBR"
##  [3] "STCNTY"
##  [4] "County"
##  [5] "FIPS"
##  [6] "County FIPS"
##  [7] "Location"
##  [8] "Total Score"
##  [9] "Max Possible Score"
## [10] "Hardest Hit Area (HHA)"
## [11] "HHA Score"
## [12] "Low Income Area (LIA) County SAIPE - (Poverty Percentage)"
## [13] "Low Income Area (LIA) County SAIPE- Score"
## [14] "Low Income Area (LIA) Census Tract (Poverty Percentage)"
## [15] "Low Income Area (LIA) Census Tract - Score"
## [16] "Tribal Community (1 if yes)"
## [17] "Tribal Community Score (Geographic Only)"
## [18] "Rural"
## [19] "Rural - Score"
```

```
#One Hot encoding:
doh<- one_hot(as.data.table(d))

d$ST_ABBR=as.factor(d$ST_ABBR)
d$`Tribal Community (1 if yes)`= as.factor(d$`Tribal Community (1 if yes)`)


dr<-(d$Rural==1)
dnr<- d[!dr,]
dr<-d[dr,]

#Preparing dataframes:
d_reg = d %>% group_by(STATE,ST_ABBR)  %>%
  summarise( HHA= mean(`HHA Score`),
             LIASAIPE = mean(`Low Income Area (LIA) County SAIPE- Score`),
             LIACT= mean(`Low Income Area (LIA) Census Tract - Score`),
             RuralScore= mean(`Rural - Score`),
             LIASAIPEperc=mean(`Low Income Area (LIA) County SAIPE - (Poverty
Percentage)`),
             LIACTperc=mean(`Low Income Area (LIA) Census Tract (Poverty
Percentage)`),
             TS=mean(`Max Possible Score`),
             .groups = 'drop')
names(d)
```

```
##  [1] "STATE"
##  [2] "ST_ABBR"
##  [3] "STCNTY"
##  [4] "County"
##  [5] "FIPS"
##  [6] "County FIPS"
##  [7] "Location"
##  [8] "Total Score"
##  [9] "Max Possible Score"
## [10] "Hardest Hit Area (HHA)"
## [11] "HHA Score"
## [12] "Low Income Area (LIA) County SAIPE - (Poverty Percentage)"
## [13] "Low Income Area (LIA) County SAIPE- Score"
## [14] "Low Income Area (LIA) Census Tract (Poverty Percentage)"
## [15] "Low Income Area (LIA) Census Tract - Score"
## [16] "Tribal Community (1 if yes)"
## [17] "Tribal Community Score (Geographic Only)"
## [18] "Rural"
## [19] "Rural - Score"
```

```
d_hha= doh %>% group_by(STATE,ST_ABBR)  %>%
  summarise( SH= sum(`Hardest Hit Area (HHA)_SustainedHotspot`),
             H = sum(`Hardest Hit Area (HHA)_Hotspot`),
             MB= sum(`Hardest Hit Area (HHA)_ModerateBurden`),
             LB= sum(`Hardest Hit Area (HHA)_LowBurden`),
             MBR=sum(`Hardest Hit Area (HHA)_ModerateBurdenResolving`),
             HBR=sum(`Hardest Hit Area (HHA)_HighBurdenResolving`),
             EH=sum(`Hardest Hit Area (HHA)_EmergingHotspot`),
             .groups = 'drop')


sapply(d_hha,  class)
```

```
##       STATE     ST_ABBR          SH           H          MB          LB
## "character" "character"   "integer"   "integer"   "integer"   "integer"
##         MBR         HBR          EH
##   "integer"   "integer"   "integer"
```

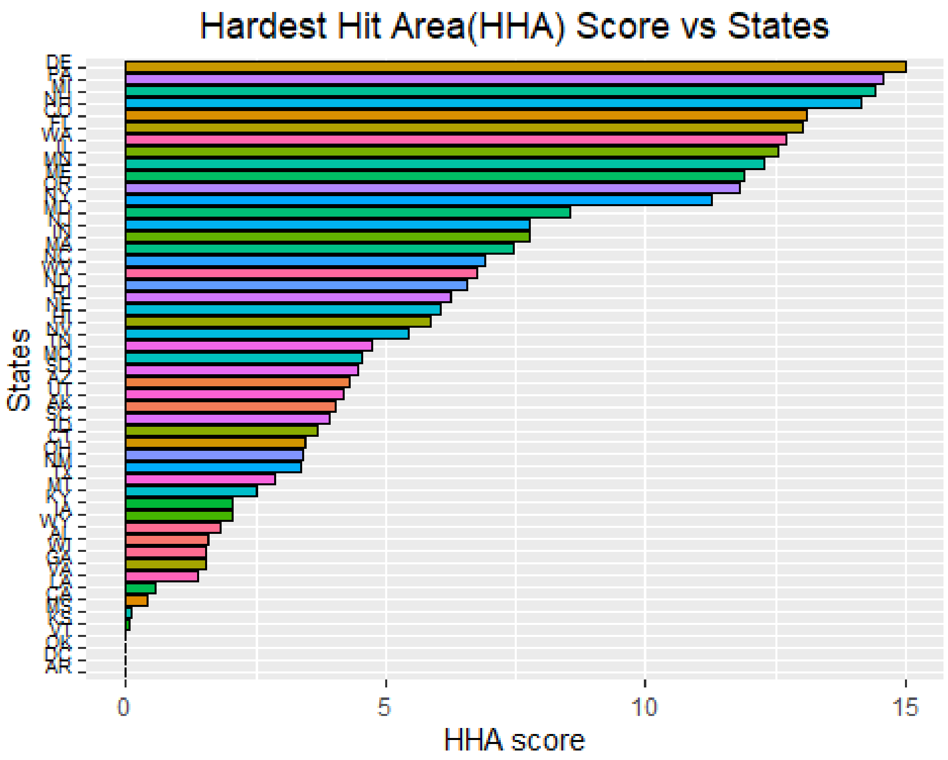
```
is.na(row.names(d_hha))
```

```
##  [1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [13] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [25] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [37] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE
FALSE
## [49] FALSE FALSE FALSE
```

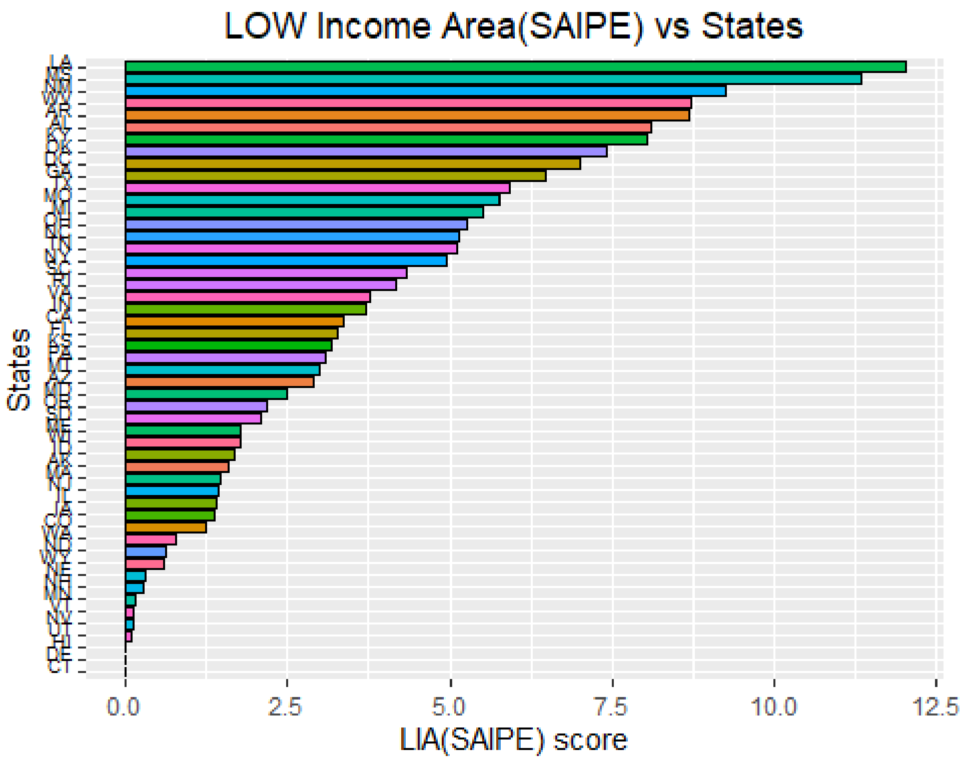
```
d_hha_tb=as.tibble(d_hha)
```

```
## Warning: `as.tibble()` was deprecated in tibble 2.0.0.
## Please use `as_tibble()` instead.
## The signature and semantics have changed, see `?as_tibble`.
```

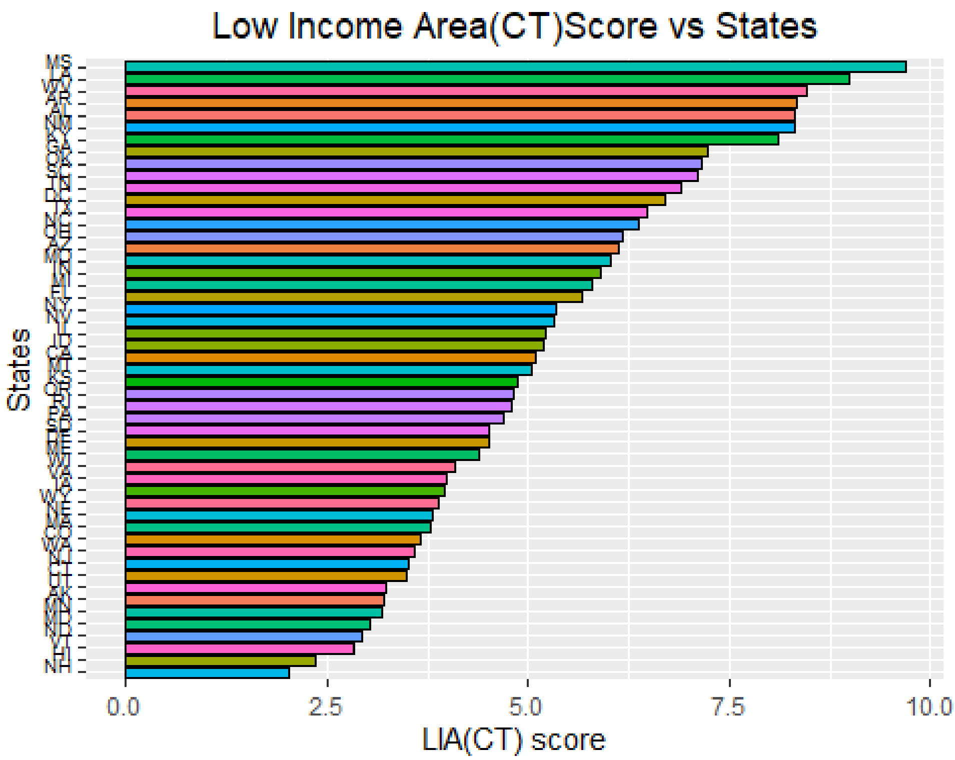
```
d_hha_tb$STATE<-tolower(d_hha_tb$STATE)
d_hha$STATE<-tolower(d_hha$STATE)
colnames(d_hha_tb)[1]= 'region'
colnames(d_hha)[1]= 'region'

#Individual Score Analysis For Every State:
ggplot(d_reg, aes(y =reorder(`ST_ABBR`, `HHA`), x=`HHA`, fill=STATE)) +
  geom_bar(stat = 'identity', color='black')+labs(title= "Hardest Hit
Area(HHA) Score vs States",
                                                  y="States", x = "HHA
score")+
  theme(plot.title = element_text(hjust = 0.5),legend.position="none",
        axis.text.y = element_text(color = "black", size =7, angle = 0, hjust
= 1,
                                   vjust = 0, face = "plain"))
```



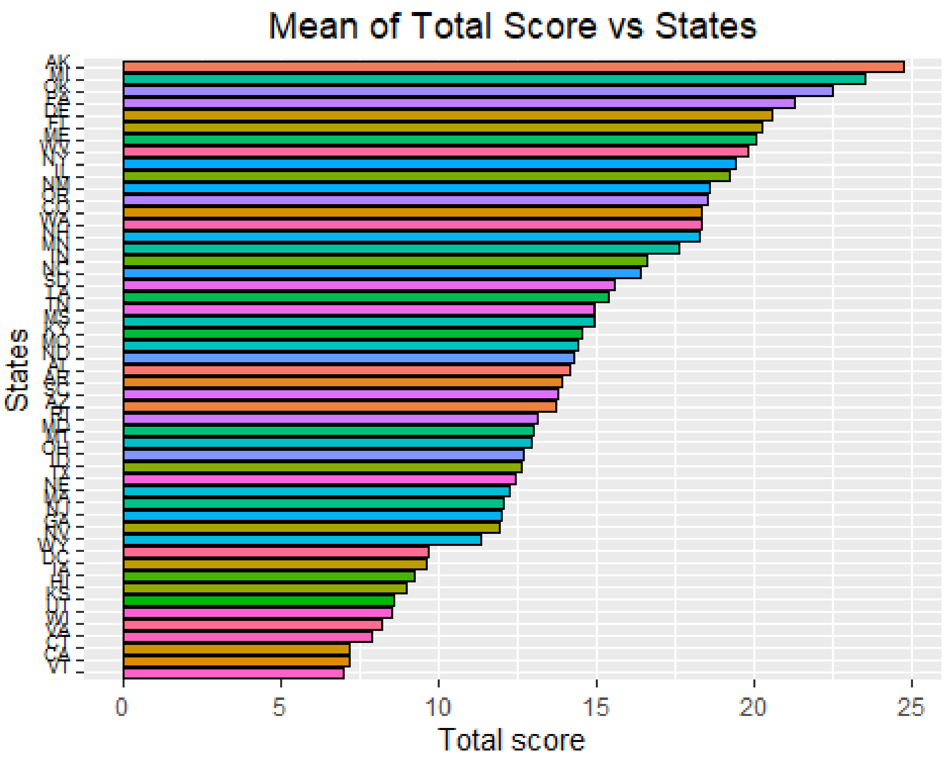
```
ggplot(d_reg, aes(y =reorder(`ST_ABBR`, `LIASAIPE`), x=`LIASAIPE`,
fill=STATE)) +
  geom_col(stat = 'identity', color='black')+labs(title= "LOW Income
Area(SAIPE) vs States",
                                                  y="States", x = "LIA(SAIPE)
score")+
  theme(plot.title = element_text(hjust = 0.5),legend.position="none",
        axis.text.y = element_text(color = "black", size =7, angle = 0, hjust
= 1,
                                   vjust = 0, face = "plain"))

## Warning: Ignoring unknown parameters: stat
```



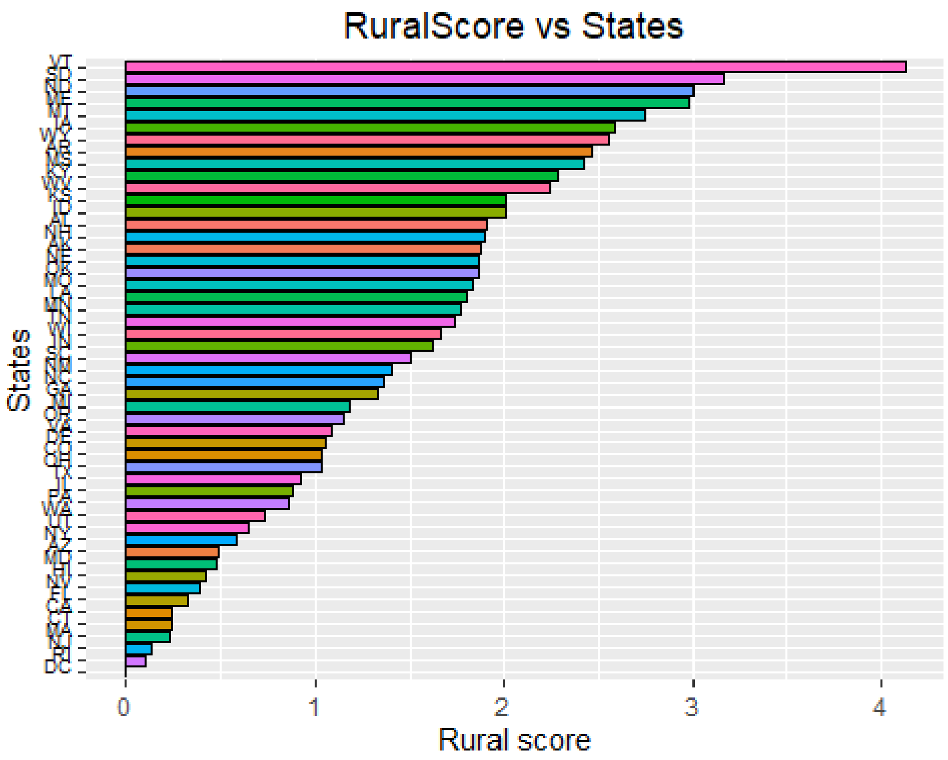
```
ggplot(d_reg, aes(y =reorder(`ST_ABBR`, `LIACT`), x=`LIACT`, fill=STATE)) +
  geom_bar(stat = 'identity', color='black')+labs(title= "Low Income
Area(CT)Score vs States",
                                                  y="States", x = "LIA(CT)
score")+
  theme(plot.title = element_text(hjust = 0.5),legend.position="none",
        axis.text.y = element_text(color = "black", size =7, angle = 0, hjust
= 1,
                                   vjust = 0, face = "plain"))
```



```
ggplot(d_reg, aes(y =reorder(`ST_ABBR`, `TS`), x=`TS`, fill=STATE)) +
  geom_bar(stat = 'identity', color='black')+labs(title= "Mean of Total Score
vs States",
                                                  y="States", x = "Total
score")+
  theme(plot.title = element_text(hjust = 0.5),legend.position="none",
        axis.text.y = element_text(color = "black", size =7, angle = 0, hjust
= 1,
                                   vjust = 0, face = "plain"))
```

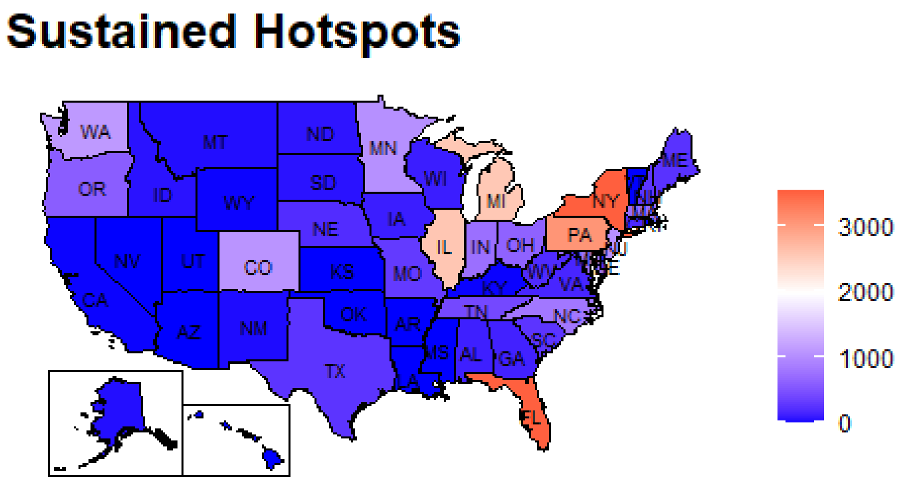


```
ggplot(d_reg, aes(y =reorder(`ST_ABBR`, `RuralScore`), x=`RuralScore`,
fill=STATE)) +
  geom_bar(stat = 'identity', color='black')+labs(title= "RuralScore vs
States",
                                                  y="States", x = "Rural
score")+
  theme(plot.title = element_text(hjust = 0.5),legend.position="none",
        axis.text.y = element_text(color = "black", size =7, angle = 0, hjust
= 1,
                                   vjust = 0, face = "plain"))
```

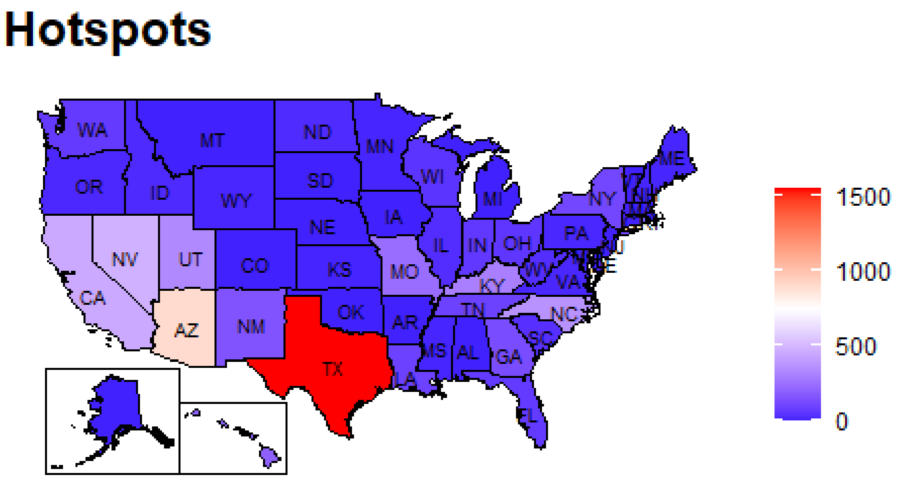


```
#Cloropleth Map:
us<-map_data("state")
dh <- us %>%
  left_join(d_hha, by='region')
dh$region<-factor(dh$region)
dh <- dh[order(dh$order),]


centroids <- data.frame(region=tolower(state.name), long=state.center$x,
lat=state.center$y)
centroids$abb<-state.abb[match(centroids$region,tolower(state.name))]
statenames<-data.frame(
  region=levels(dh$region)
)
# Merge it with centroids
centroids<-merge(statenames,centroids,by="region")

p1 <- ggplot(d_hha, aes(map_id = region)) +
  geom_map(aes(fill = SH), map = fifty_states, color='black')+
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map()+
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 2000)+
  theme_minimal()+
  labs(
    title = 'Sustained Hotspots',
    x="", y="", fill=""
  )+
  theme(
    plot.title = element_text(size=18, face='bold', color = 'black')
  )+
  with(centroids,
       annotate(geom="text", x = long, y=lat, label = abb,
                size = 2.5,color="black",family="Times")
  )+
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(panel.background = element_blank()) +
  fifty_states_inset_boxes()
p1
```

```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y,
:
## font family not found in Windows font database
```

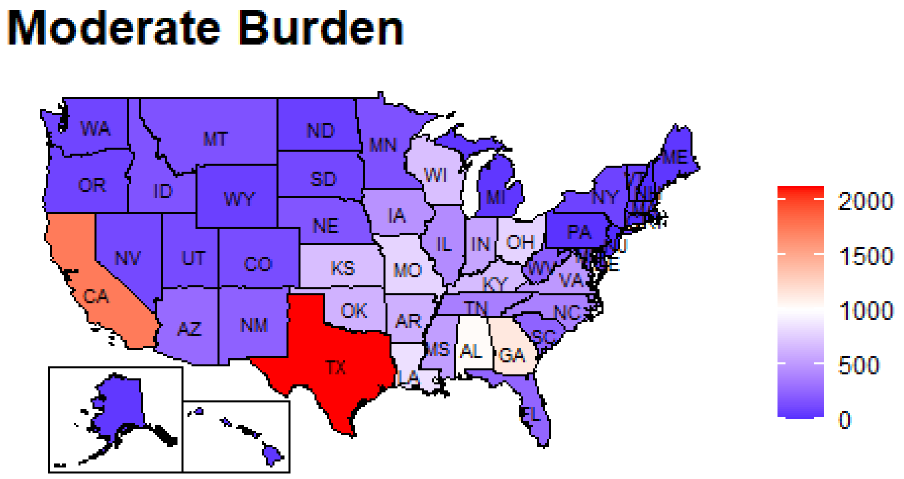


```
p2 <- ggplot(d_hha, aes(map_id = region)) +
  geom_map(aes(fill = H), map = fifty_states, color='black')+
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map()+
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 750)+
  theme_minimal()+
  labs(
    title = 'Hotspots',
    x="", y="", fill=""
  )+
  theme(
    plot.title = element_text(size=18, face='bold', color = 'black')
  )+
  with(centroids,
       annotate(geom="text", x = long, y=lat, label = abb,
                size = 2.5,color="black",family="Times")
  )+
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(panel.background = element_blank()) +
  fifty_states_inset_boxes()
p2

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y,
:
## font family not found in Windows font database
```

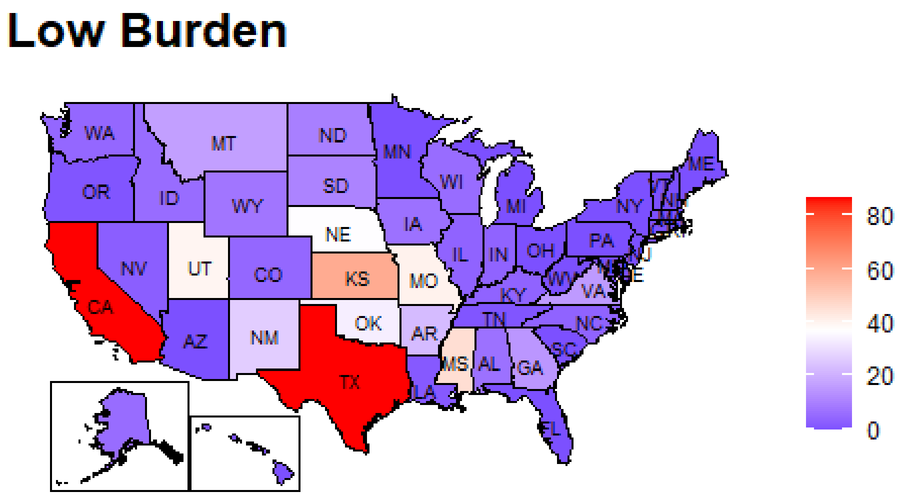


```
p3 <- ggplot(d_hha, aes(map_id = region)) +
  geom_map(aes(fill = MB), map = fifty_states, color='black')+
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map()+
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 1000)+
  theme_minimal()+
  labs(
    title = 'Moderate Burden',
    x="", y="", fill=""
  )+
  theme(
    plot.title = element_text(size=18, face='bold', color = 'black')
  )+
  with(centroids,
       annotate(geom="text", x = long, y=lat, label = abb,
                size = 2.5,color="black",family="Times")
  )+
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(panel.background = element_blank()) +
  fifty_states_inset_boxes()
p3
```

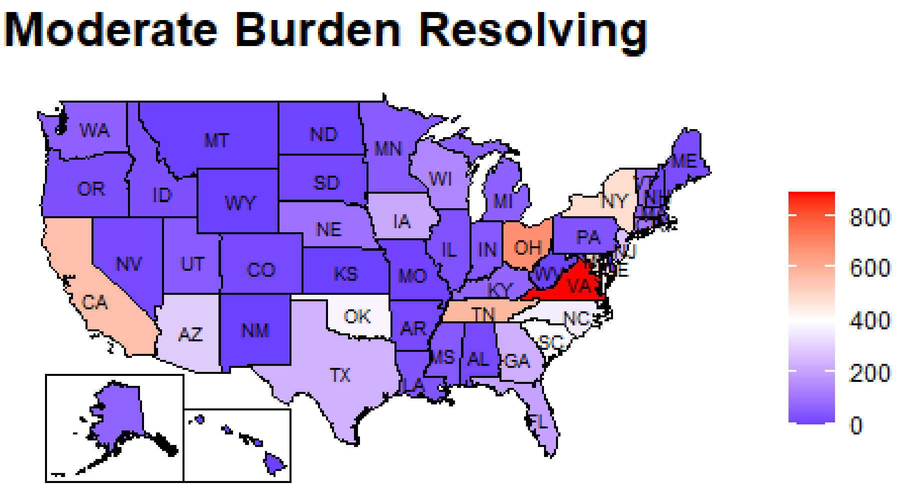
```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y,
:
## font family not found in Windows font database
```



```
p4 <- ggplot(d_hha, aes(map_id = region)) +
  geom_map(aes(fill = LB), map = fifty_states, color='black')+
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map()+
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 37.5)+
  theme_minimal()+
  labs(
    title = 'Low Burden',
    x="", y="", fill=""
  )+
  theme(
    plot.title = element_text(size=18, face='bold', color = 'black')
  )+
  with(centroids,
       annotate(geom="text", x = long, y=lat, label = abb,
                size = 2.5,color="black",family="Times")
  )+
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(panel.background = element_blank()) +
  fifty_states_inset_boxes()
p4

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y,
:
## font family not found in Windows font database
```

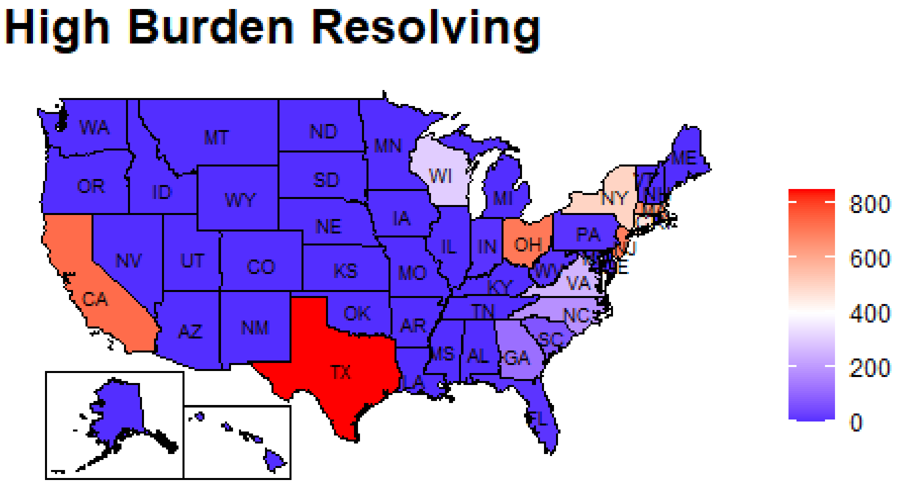


```
p5 <- ggplot(d_hha, aes(map_id = region)) +
  geom_map(aes(fill = MBR), map = fifty_states, color='black')+
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map()+
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 400)+
  theme_minimal()+
  labs(
    title = 'Moderate Burden Resolving',
    x="", y="", fill=""
  )+
  theme(
    plot.title = element_text(size=18, face='bold', color = 'black')
  )+
  with(centroids,
       annotate(geom="text", x = long, y=lat, label = abb,
                size = 2.5,color="black",family="Times")
  )+
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(panel.background = element_blank()) +
  fifty_states_inset_boxes()
p5

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y,
:
## font family not found in Windows font database
```

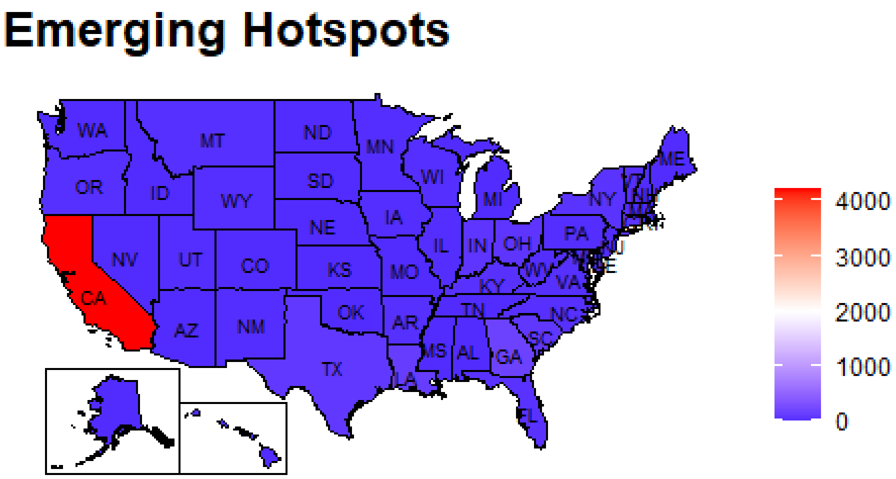


```r
p6 <- ggplot(d_hha, aes(map_id = region)) +
  geom_map(aes(fill = HBR), map = fifty_states, color='black')+
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map()+
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 400)+
  theme_minimal()+
  labs(
    title = 'High Burden Resolving',
    x="", y="", fill=""
  )+
  theme(
    plot.title = element_text(size=18, face='bold', color = 'black')
  )+
  with(centroids,
       annotate(geom="text", x = long, y=lat, label = abb,
                size = 2.5,color="black",family="Times")
  )+
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(panel.background = element_blank()) +
  fifty_states_inset_boxes()
p6
```

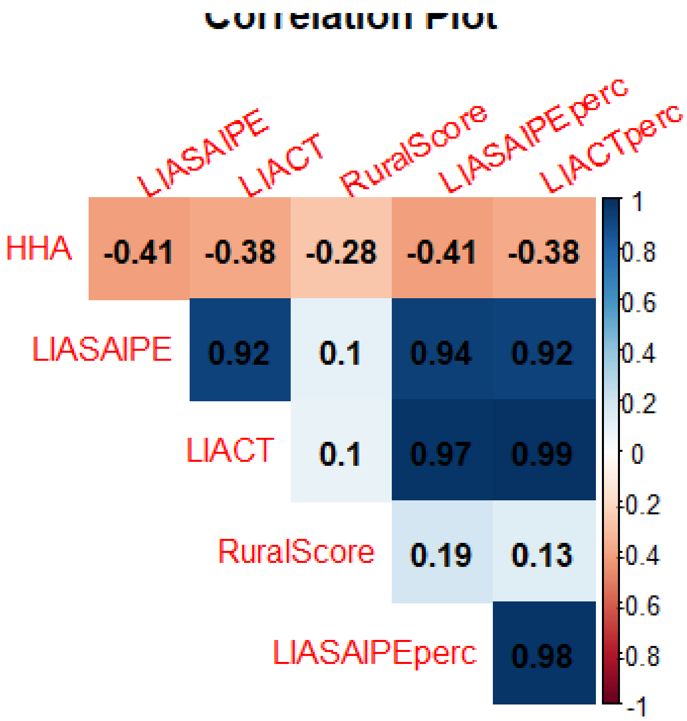
```
## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y,
:
## font family not found in Windows font database
```



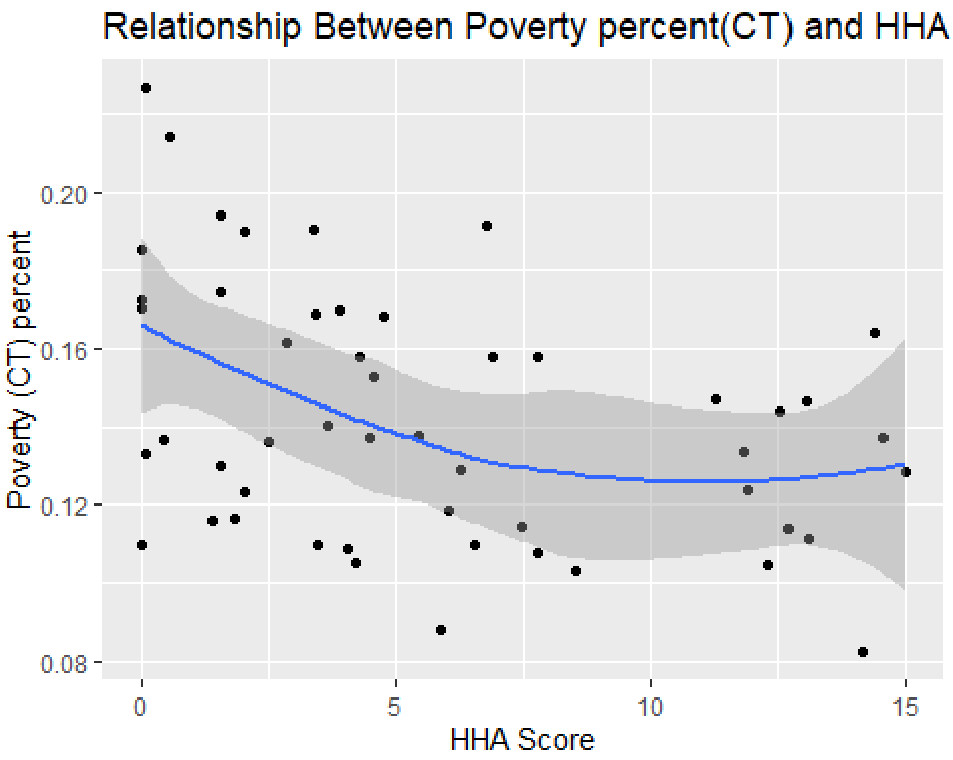
```
p7 <- ggplot(d_hha, aes(map_id = region)) +
  geom_map(aes(fill = EH), map = fifty_states, color='black')+
  expand_limits(x = fifty_states$long, y = fifty_states$lat) +
  coord_map()+
  scale_fill_gradient2(low = "blue", mid = "white", high = "red",
                       midpoint = 2000)+
  theme_minimal()+
  labs(
    title = 'Emerging Hotspots',
    x="", y="", fill=""
  )+
  theme(
    plot.title = element_text(size=18, face='bold', color = 'black')
  )+
  with(centroids,
       annotate(geom="text", x = long, y=lat, label = abb,
                size = 2.5,color="black",family="Times")
  )+
  scale_x_continuous(breaks = NULL) +
  scale_y_continuous(breaks = NULL) +
  labs(x = "", y = "") +
  theme(panel.background = element_blank()) +
  fifty_states_inset_boxes()
p7

## Warning in grid.Call.graphics(C_text, as.graphicsAnnot(x$label), x$x, x$y,
:
## font family not found in Windows font database
```



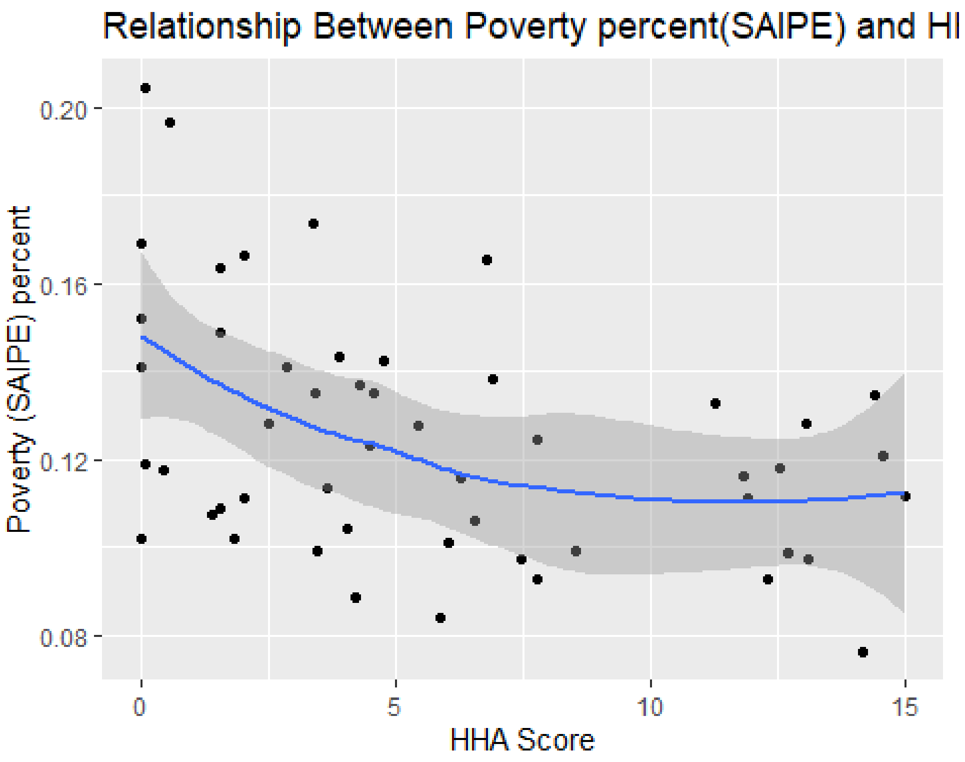
```
#Correlation Plot:
vars <- dplyr::select(d_reg,`HHA`, everything(), -c(STATE,ST_ABBR, TS))
corrplot(cor(vars[sapply(vars, function(x) !is.factor(x))]),type="upper",
method="color", diag=FALSE,
        tl.srt=30, addCoef.col="black", main="Correlation Plot")
```



```
#Relationship Between Poverty percent and HHA scores:
ggplot(d_reg, aes(x=`HHA` ,y=`LIACTperc`)) +
  geom_point()+geom_smooth()+
  labs(x="HHA Score",
       y="Poverty (CT) percent",
       title="Relationship Between Poverty percent(CT) and HHA scores")

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```



```
#Relationship Between Poverty percent and HHA scores:
ggplot(d_reg, aes(x=`HHA` ,y=`LIASAIPEperc`)) +
  geom_point()+geom_smooth()+
  labs(x="HHA Score",
       y="Poverty (SAIPE) percent",
       title="Relationship Between Poverty percent(SAIPE) and HHA scores")

## `geom_smooth()` using method = 'loess' and formula 'y ~ x'
```


Relationship Between Poverty percent(SAIPE) and HI
Poverty (SAIPE) percent
0.20
0.16
0.12
0.08
0
5
10
15
HHA Score