SDGB 7844 HW 2: Townsend Material Deprivation Index

Jiayin Hu

2018/10/18

Before the questions and answers, we list the R packages we need in this homework report.

```r
#setting the packages needed
require(tidyverse)
require(knitr)
require(gridExtra)
require(rgdal)
require(RColorBrewer)
require(GGally)
```

Q1

What is a census tract? How many census tracts are in New York County? (Provide the citations for references used.)

Census Tracts are small, relatively permanent statistical subdivisions of a county or equivalent entity that are updated by local participants prior to each decennial census as part of the Census Bureau's Participant Statistical Areas Program.[1]

There are 288 census tracts in New York County.[2]

Q2

Describe one advantage and one disadvantage of computing estimates after combining 5-years of data.

Advantage: The estimates after combining 5-year data are based on larger sample sizes so that the results are more reliable. Some places with small population which do not meet the particular population threshold do not publish the data every year. So the 5-year data is more comprehensive.

[1] https://www.census.gov/geo/reference/gtc/gtc_ct.html

[2] https://www2.census.gov/geo/maps/dc10map/tract/st36_ny/c36061_new_york/DC10CT_C36061_CT2MS.txt

Disadvantage: Some statistics in some places may change dramatically in the more recent period, so it is possible that the estimates based on 5-year data provide less current information.

Q3

Download the ACS data for 5-year estimates spanning from 2011-2015. (This means 2015 5-year estimates, not a combination of 1-year estimates for 2011, 1-year estimates for 2012, etc.) for for all New York County census tracts for the following variables using American FactFinder (link; further instructions can be found on Blackboard in the file "Downloading Map and ACS Data"). Table DP03 contains selected economic characteristics and Table DP04 includes selected housing characteristics. Each row in the table represents a single census tract in New York County.

- unemployment: Table DP03, variable HC03 VC07

- housing tenure (whether house is rented or owned): Table DP04, variable HC03 VC66
- no vehicles: Table DP04, variable HC03 VC85
- low occupancy: Table DP04, variable HC03 VC113 (You will have to transform this variable to get % overcrowded to use in the index)

Clean the data and merge the tables into one data frame, each row representing a census tract, each column representing one of the Townsend variables (keep the geography columns).

For each variable, construct a histogram and compute the following summary statistics: mean, median, standard deviation, maximum, and minimum. Describe the shape of each histogram.

```
#import data
dp03 <- read_delim("ACS_15_5YR_DP03/ACS_15_5YR_DP03_with_ann.csv",
                   delim = ",", col_name = TRUE)

## Parsed with column specification:
## cols(
##   .default = col_character()
## )

## See spec(...) for full column specifications.

dp04 <- read_delim("ACS_15_5YR_DP04/ACS_15_5YR_DP04_with_ann.csv",
                   delim = ",", col_name = TRUE)

## Parsed with column specification:
## cols(
##   .default = col_character()
## )
## See spec(...) for full column specifications.
```

```
#creat a tible using target statistics
#delete the description
#convert the type of the number to numeric (missing values convert to NA)
#rename columns for ease of use
townsend <- tibble(geo = dp03$'GEO.id2'[-1],
                   unemp = as.numeric(dp03$HC03_VC07[-1]),
                   rent = as.numeric(dp04$HC03_VC66[-1]),
                   car = as.numeric(dp04$HC03_VC85[-1]),
                   oc = 100 - as.numeric(dp04$HC03_VC113[-1]))
```

Show the tibble we created.

```
townsend

## # A tibble: 288 x 5
##    geo         unemp  rent   car    oc
##    <chr>       <dbl> <dbl> <dbl> <dbl>
##  1 36061000100  NA    NA    NA    NA
##  2 36061000201   1.2  99.4  90    11.6
##  3 36061000202   7.3  75.8  73.9   2.7
##  4 36061000500  NA    NA    NA    NA
##  5 36061000600   3.8  97.2  75.5   9.4
##  6 36061000700   3.8  80.1  87    11.6
##  7 36061000800   5.7  98.4  83.2 14
##  8 36061000900   2.3  89.2  76.4   2.80
##  9 36061001001   7.3  22.6  51.5   2.2
## 10 36061001002   8.7 100    87.3   2.80
## # ... with 278 more rows

mean <- apply(townsend[2:5], 2, mean, na.rm = TRUE)
median <- apply(townsend[2:5], 2, median, na.rm = TRUE)
std <- apply(townsend[2:5], 2, sd, na.rm = TRUE)
max <- apply(townsend[2:5], 2, max, na.rm = TRUE)
min <- apply(townsend[2:5], 2, min, na.rm = TRUE)
summaryTable <- data.frame(mean, median, std, max, min)
kable(summaryTable, caption="Summary Statistics of Each Variable", col.names
= c("Mean", "Median", "Std", "Max", "Min"), align = "c")
```

Summary Statistics of Each Variable

	Mean	Median	Std	Max	Min
unemp	4.926855	4.00	3.250082	25.0	0
rent	77.531429	80.75	19.173203	100.0	0
car	77.413571	78.60	9.783584	100.0	0
oc	5.883214	4.10	5.164891	36.4	0

Then, plot the histogram.

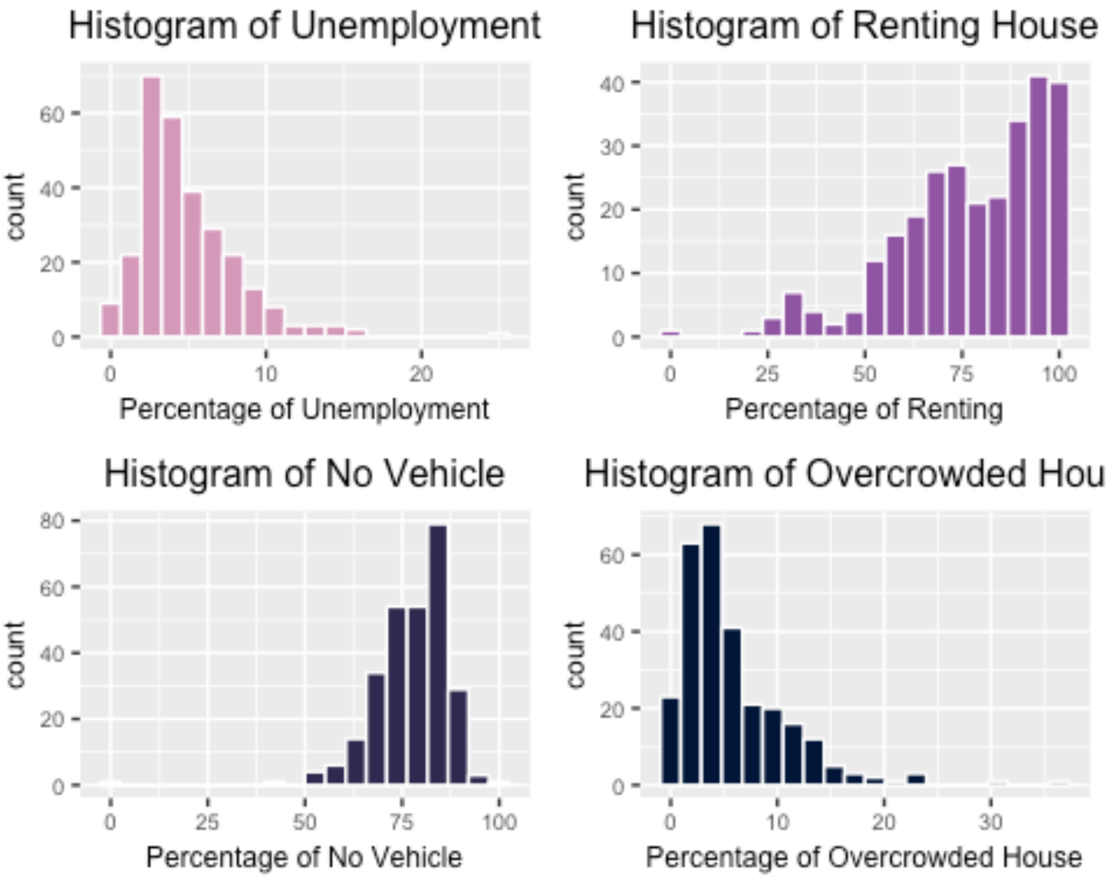
```
theme.info <- theme(plot.title = element_text(size = 12, hjust= 0.5),
                    axis.title = element_text(size = 9),
                    axis.text = element_text(size = 7))
#Plot histograms of the variables
unempPlot <- ggplot(townsend, aes(unemp)) + geom_histogram(bins = 20, color =
"white", fill = "#D499B9") +
  ggtitle("Histogram of Unemployment") +
  labs(x="Percentage of Unemployment") + theme.info


rentPlot <- ggplot(townsend, aes(rent)) + geom_histogram(bins = 20, color =
"white", fill = "#9055A2") +
  ggtitle("Histogram of Renting House") +
  labs(x="Percentage of Renting") + theme.info


carPlot <- ggplot(townsend, aes(car)) + geom_histogram(bins = 20, color =
"white", fill = "#2E294E") +
  ggtitle("Histogram of No Vehicle") +
  labs(x="Percentage of No Vehicle") + theme.info


ocPlot <- ggplot(townsend, aes(oc)) + geom_histogram(bins = 20, color =
"white", fill = "#011638") +
  ggtitle("Histogram of Overcrowded House") +
  labs(x="Percentage of Overcrowded House") + theme.info


#arrange the histogram in one plot
plot = grid.arrange(unempPlot, rentPlot, carPlot, ocPlot, ncol = 2, nrow = 2)
```



The histograms of percentage of unemployment and overcrowded house are right skewed. The histograms of percentage of renting house and no vehicle are left skewed.

Q4

How many observations are missing for each variable? What percentage of census tracts do not have complete data? Is this a problem for our analysis? Justify your answer. (Note: do not delete tracts with missing data.)

```
#count NA in each columns
apply(townsend[,c("unemp", "rent", "car", "oc")], 2,
function(x){sum(is.na(x))})

## unemp  rent   car    oc
##     5     8     8     8
```

The numbers of missing value in variable unemployment, rent, car and occupancy are 5, 8, 8, 8, respectively.

```
#percentage of census tracts do not have complete data
length(which(!complete.cases(townsend)))/nrow(townsend)

## [1] 0.02777778
```

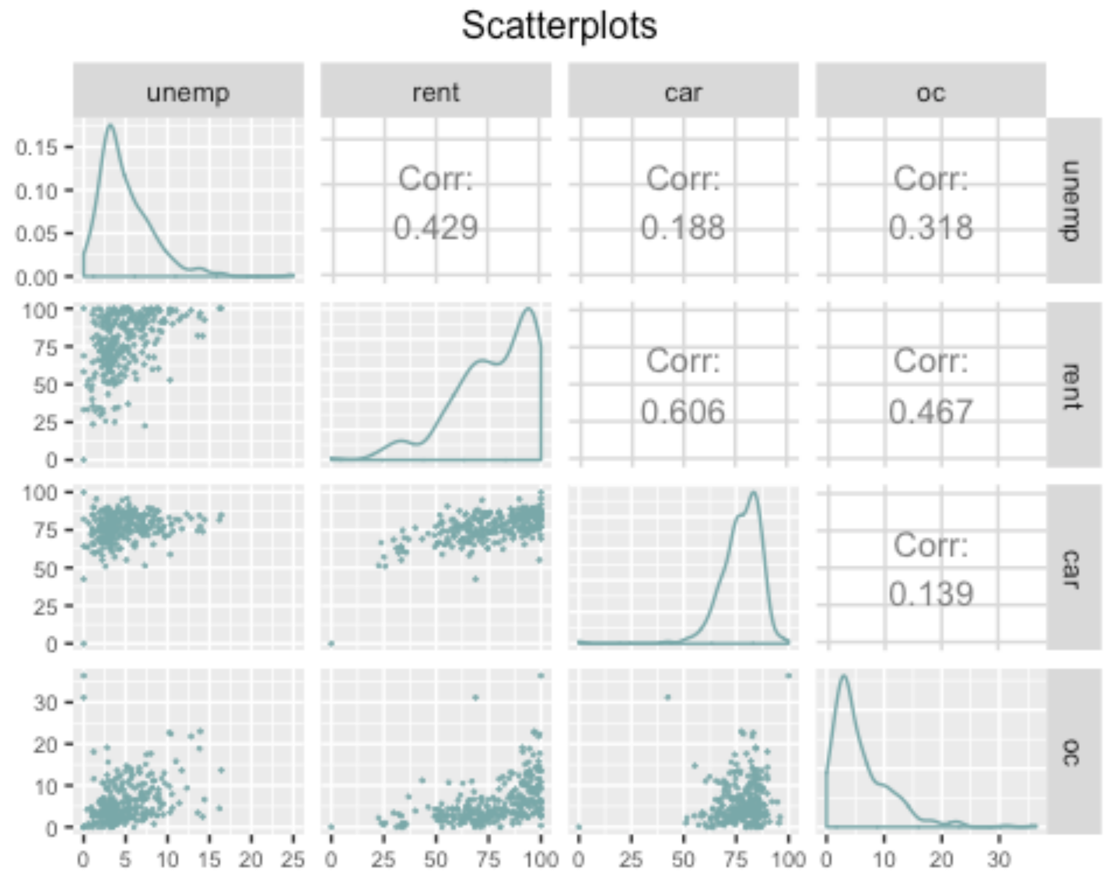
About 2.78% of census tracts do not have complete data. This will not cause big problems to our analysis. Because the proportion of missing values are relatively small and our analysis next is unecessary to use perfectly complete data. So we can just ignore the NA value.

Q5

Construct scatterplots of the four variables. Are they linearly related? Now, transform the variables as given in step (a), adding the transformed variables to your data frame. Make another scatter plot matrix with the transformed variables. Are they linearly related? Construct a correlation matrix of the transformed variables and describe your results.

Plot the scatterplots by `ggpairs()`.

```r
#use ggpairs from GGally package to plot scatter matrix
#also plot distribution of each variable in the diagonal plots
ggpairs(townsend[2:5],
        lower=list(continuous=wrap("points", colour="#79a8a9", size = 0.2,
alpha = 0.8)) ,
        diag = list(continuous = wrap("densityDiag", colour="#79a8a9"))) +
ggtitle("Scatterplots") + theme.info
```



We can find that the variables are not linear related.

Then we calculate the trasformation of each variable and correlation.

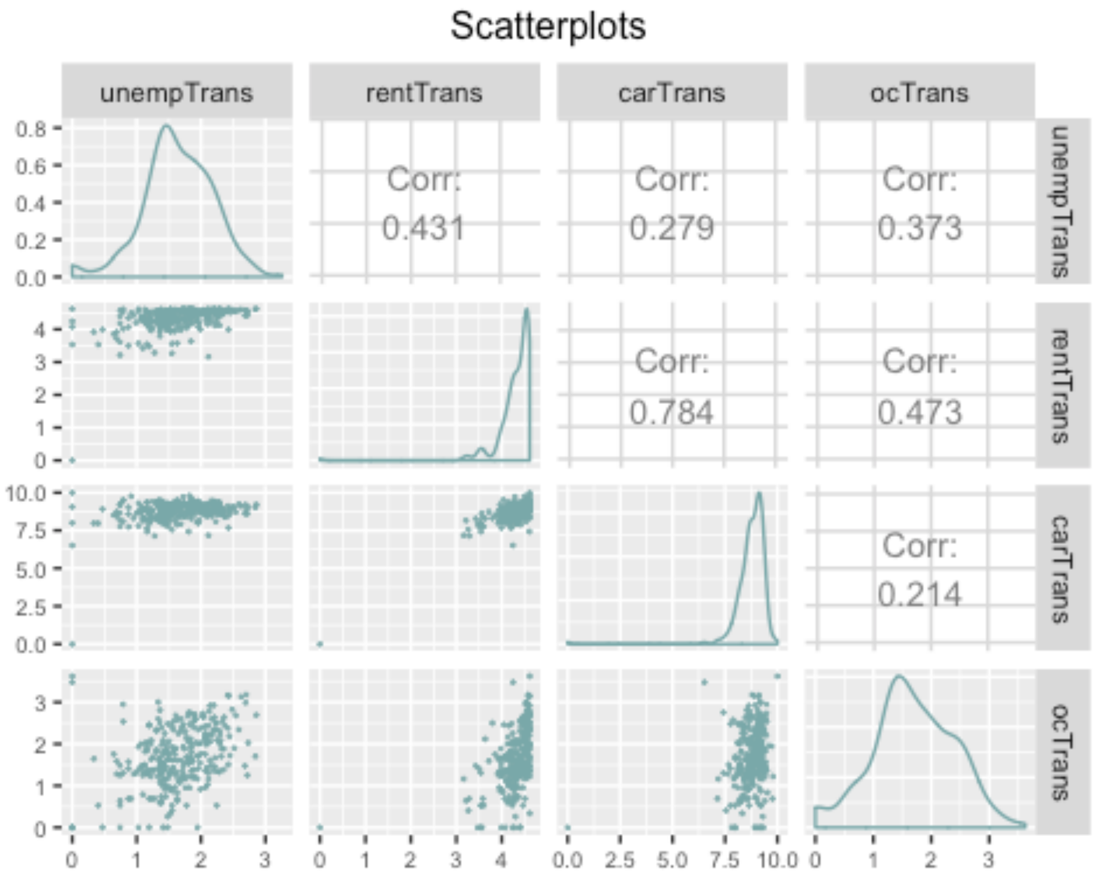
```
#transformation
unempTrans = log(townsend$unemp + 1)
ocTrans = log(townsend$oc + 1)
rentTrans = log(townsend$rent + 1)
carTrans = sqrt(townsend$car)

#add the variables after transformation into the dataframe
townsend = townsend %>% mutate(unempTrans, rentTrans, carTrans, ocTrans)

#calculate the correlation matrix
cor(townsend[6:9], use = "complete.obs")

##             unempTrans rentTrans  carTrans   ocTrans
## unempTrans  1.0000000 0.4311138 0.2788261 0.3725944
## rentTrans   0.4311138 1.0000000 0.7842310 0.4730620
## carTrans    0.2788261 0.7842310 1.0000000 0.2141658
## ocTrans     0.3725944 0.4730620 0.2141658 1.0000000

ggpairs(townsend[6:9],
        lower=list(continuous=wrap("points", colour="#79a8a9", size = 0.2,
alpha = 0.8)) ,
        diag = list(continuous = wrap("densityDiag", colour="#79a8a9"))) +
ggtitle("Scatterplots") + theme.info
```



From the plots, the varibles are still not strongly linear related. Only the correlation between `rentTrans` and `carTrans` is relatively higher, but the scatterplot of it is so concentrated which does not contain much information. The correlation of other variables are low. So they are not lineae related.

Q6

Compute the Townsend index value for each census tract. Add the index to your data frame. For how many census tracts are you able to compute the Townsend index? Why does this number not equal the total number of census tracts?

```
#compute Townsend index value
meanTrans = apply(townsend[6:9], 2, mean, na.rm = TRUE)
stdTrans = apply(townsend[6:9], 2, sd, na.rm = TRUE)
z = data.frame((townsend[6:9][,1] - meanTrans[1])/stdTrans[1],
               (townsend[6:9][,2] - meanTrans[2])/stdTrans[2],
               (townsend[6:9][,3] - meanTrans[3])/stdTrans[3],
               (townsend[6:9][,4] - meanTrans[4])/stdTrans[4])
```

```
townsendIndex = apply(z, 1, sum)
townsend["Townsend Index"] = townsendIndex
townsend["Townsend Index"]
```

```
## # A tibble: 288 x 1
##    `Townsend Index`
##               <dbl>
##  1           NA
##  2            1.36
##  3            0.179
##  4           NA
##  5            1.38
##  6            2.03
##  7            3.13
##  8           -0.873
##  9           -5.02
## 10            2.24
## # ... with 278 more rows
```

```
sum(is.na(townsend$"Townsend Index")!=TRUE)
```

```
## [1] 280
```

There are 280 indice in our result. This does not match the number of census tracts because there are 8 incomplete rows containing missing data. We could not calculate the Townsend Index for them.

Q7

Identify which census tract is the most deprived and which is the least deprived (give the census tract number and deprivation index level). Based on your results, would you like to live in the least deprived census tract? Justify your answer.

```
#identify deprived level
townsend[which.max(townsend$`Townsend Index`), c(1, 2, 3, 4, 5, 10)] #Most
deprived area
```

```
## # A tibble: 1 x 6
##   geo         unemp  rent   car    oc `Townsend Index`
##   <chr>       <dbl> <dbl> <dbl> <dbl>            <dbl>
## 1 36061028500  16.4   100    85  13.7             5.04
```

```
townsend[which.min(townsend$`Townsend Index`), c(1, 2, 3, 4, 5, 10)] #Lease
deprived area
```

```
## # A tibble: 1 x 6
##   geo         unemp  rent   car    oc `Townsend Index`
##   <chr>       <dbl> <dbl> <dbl> <dbl>            <dbl>
## 1 36061021703     0     0     0     0            -28.6
```

The most deprived area is the tract whose GEOID is 36061028500 with Townsend Index 5.044114.
The lease deprived area is the tract whose GEOID is 36061021703 with Townsend Index -28.58582.
Personally speaking, I would not like to live in the least deprived census tract. Because this census tract is located in the Upper Manhattan which is far from my school and more interesting places in the city.

Q8

The ACS data includes not only estimates but their margins of error which we ignored in our calculations. What are the implications?

The estimate is calculate from the sample which means it may vary from the real value. So the margin of error measures the possible variation of an estimate around the real value. Therefore, the estimate may not correspond to the real value perfectly. The margin of error, however, provides the estimate a given level of confidence.
We ignore it because we hold the specific confidence that the estimate is not much different from the real value.

Q9

Construct a map color-coded by the deprivation index value quintile. Each quintile (i.e., 20%) should be assigned a different color from least to most deprived. Download the shape files for New York state census tracts for 2015 from the U.S. Census Bureau website (link; further instructions can be found on Blackboard in the file "Downloading Map and ACS Data"). Extract the tracts for New York County only. Include a legend and plot title. Describe the patterns you see, especially in relation to what you know about neighborhoods in New York City. What does the large rectangle in the middle of the map represent?

```r
nyMap <- readOGR(dsn = "tl_2015_36_tract", layer = "tl_2015_36_tract")
```

```
## OGR data source with driver: ESRI Shapefile
## Source: "/Users/amo-yinyin/Desktop/2018FALLCourses/STAT METHODS AND COMP I/Homework/HW2/tl_2015_36_tract", layer: "tl_2015_36_tract"
## with 4918 features
## It has 12 fields
## Integer64 fields read as strings:  ALAND AWATER
```

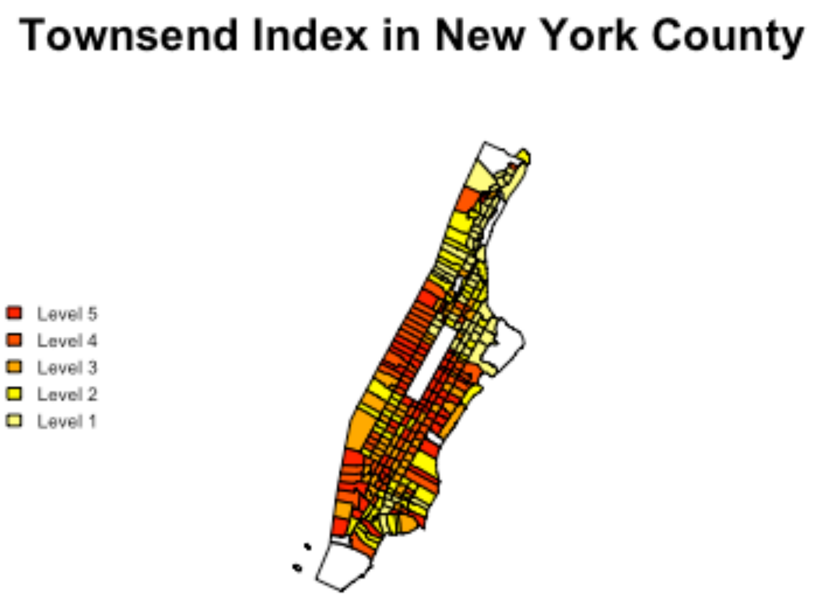
```r
nyMap <- subset(nyMap, is.element(nyMap$GEOID, townsend$geo))

townsendmap <- townsend %>% mutate(group =
                                cut(`Townsend Index`,
                                breaks=quantile(townsend$`Townsend Index`,
                                                na.rm = TRUE, probs =
                                                  seq(0, 1, 0.2))))
```

```r
color.palette <- heat.colors(n=length(levels(townsendmap$group)))
townsendmap <- townsendmap %>% mutate(color = cut(`Townsend Index`, breaks =
quantile(townsendmap$`Townsend Index`, na.rm = TRUE, probs = seq(0, 1, 0.2)),
labels=color.palette), "Deprivation Level" = cut(`Townsend Index`, breaks =
quantile(townsendmap$`Townsend Index`, na.rm = TRUE, probs = seq(0, 1, 0.2)),
labels=c("Level 5", "Level 4", "Level 3", "Level 2", "Level 1")))

townsendmap <- townsendmap %>% slice(match(nyMap$GEOID, geo))
plot(nyMap, col=as.character(townsendmap$color), main = "Townsend Index in
New York County")

legend("left", legend=c(levels(townsendmap$"Deprivation Level")),
       fill=c(color.palette, "white"), cex=0.5, bty="n", y.intersp=1.2,
ncol=1)
```



The deeper red color concentrates in the two sides of Central Park, and the west part of Lower Manhattan. Obviously, Upper East Side and Upper West are less deprived. People living there are rich. People living in Soho and Tribeca are also earn more.

In Midtown, the east part is less deprived, while the west part is more deprived.

In contrary, the lighter yellow distributes most in the Upper Manhatan where people earn less money.

The large rectangle in the middle of the map represents Central Park. Nobody lives there.

10. In which census tract is 140 W. 62nd St. (where we have class)? What is the deprivation level rank (where a rank of 1 is the most deprived)? Mark it on the map (use the computer, not by hand) and add it to your legend. (Provides citations for references.)

140 W. 62nd St. is in Census Tract 145 and its GEOID is 36061014500. The coordinate is (-73.9842727, 40.7698210).[3]

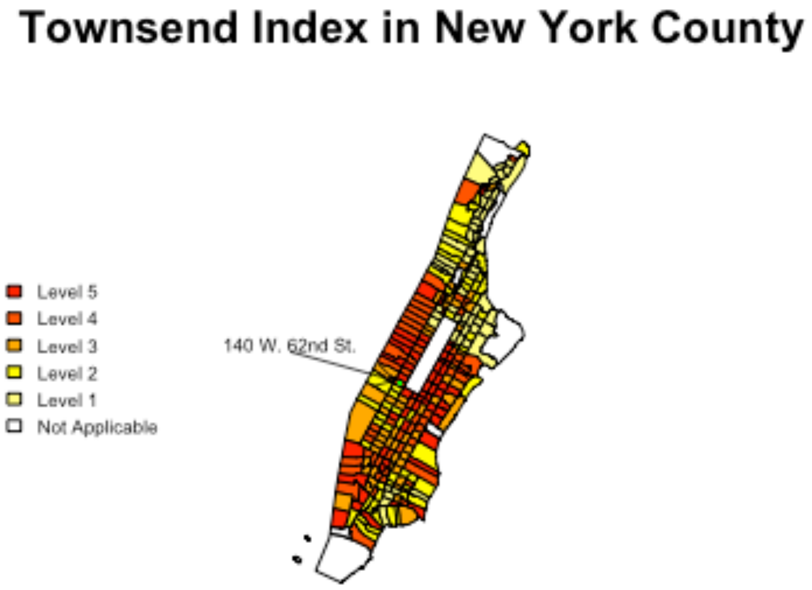
```
townsendrank <- townsend %>% mutate(Rank = rank(-townsend$"Townsend Index"))
townsendrank[which(townsendrank$geo=="36061014500"), c(1, 10, 11)]

## # A tibble: 1 x 3
##   geo         `Townsend Index`  Rank
##   <chr>                  <dbl> <dbl>
## 1 36061014500            -1.59   218
```

Its deprivation rank is 218.

```
plot(nyMap, col=as.character(townsendmap$color), main = "Townsend Index in
New York County")
legend("left", legend=c(levels(townsendmap$"Deprivation Level"), "Not
Applicable"),
       fill=c(color.palette, "white"), cex=0.5, bty="n", y.intersp=1.2,
ncol=1)
points(-73.9842727, 40.7698210, cex=0.1, pch=19, col = "green")
arrows(x0=-74.05, y0=40.78404, x1=-73.98733, y1=40.77095, length=0.1,
lwd=0.8)
text(-74.05, 40.787, labels="140 W. 62nd St.", cex = 0.5)
```

[3] https://geocoding.geo.census.gov/geocoder/geographies/onelineaddress?address=140+West+62nd+Street%2C+New+York&benchmark=4&vintage=4



11. New York County is an urban county, however New York state has roughly 22 counties classified as rural (e.g., Allegany, Essex, Otsego, Sullivan). Would it make sense to compute the Townsend index values for all census tracts within New York state combined? Why or why not?

It does not make sense to compute the Townsend index values for all census tracts within New York state combined.
Because the living condition and cost in each part of New York state may vary significantly. For example, most people living in big city, like New York City, are likely to rent house and tend to use public transportation because of the higher cost and convinient transportation. However, people living in other part of New York state may build bigger houses and drive their own cars. It tends to get a lower Townsend index value in this area, but does not mean lower level of deprivation actually.