Table of Content:

Quick Note:

On the Github you can find three different notebook and two dataset. In this manual we are gonna explain what part of the analysis had been made in each notebook and how one can follow through the codes.

There are two datasets in this study which were calculated and obtained by calculation using GIS and aggregating datasets from data.cityofchicago.org

As uploaded on the Github, “data2.csv” corresponds to redlining and canopy information regarding each neighborhoods in the map.

“dataU.csv” corresponds to other features we gathered from different sources corresponding to Household income, Electricity consumption per house,…

“Suicide.csv” corresponds to a dataset taken from data.cityofchicago.org which has a basic statistics regarding the accumulated number of suicide deaths given in a neighborhood and have some density measures as well. We used this dataset to create a suicide label (0,1,2,3) which tells us the severity of suicide rates given a neighborhood (0 is the lowest suicide rate while 3 is the highest one)

You can easily run each notebook locally by downloading the datasets and changing the path of the file in each notebook to reproduce the results.Obviously, the clusterings should be applied with the same number of clusters to get the same result as in the report and paper.

Pixels.ipynb:

In this notebook you will find the function that takes the .png file as input and compute of area covered by tree (canopy) in the given neighborhood divided by the total area of the image, i.e the total area of the neighborhood . It's important to work with the .PNG files instead of JPG since the boundary of the image collides with white blank surface of images in the JPG format whereas in the PNG images the boundary of the image taken is separable comparing to the surface.

It's worth noting that this method is an approximation of the tree canopy ratio of a given neighborhood by image. It's the best approximation using pixel counts and although it might not represent the true ratio of tree-covered land but it can be trusted as an approximation.

CanopyAnalyze.ipynb:

First, in order to load and run the notebook locally, remember to change the path to the datasets accordingly

In this notebook, we went through the basic statistics including mean, median, standard deviation and 25,50,75 percent percentile for each feature and plot the distribution (histogram) of each feature independently. (1-15)

The correlation matrix among the dataset features can be found where our main variable of interest (Canopy) had been plotted against the highest correlation (“Yellowlined” and “Redlined”) given by the matrix (15-20)

We applied the PCA method to the main dataset to reduce the dimensionality to 2 so we can plot them. The result were illustrated as the scatter plots were clustering took place on the dataset after applying PCA using 4 clusters and had been plotted in a 2-D plane using different colors representing different clusters.

We also can see in this part of the code, the elbow rule had been leveraged to determine the best number of clusters and using sum of squared errors, we see that k=4 is the best number of clusters.

After the initial clustering , we see some basic statistics (mean, median, std and percentiles) of each clusters themselves.

Clustering_ML.ipynb:

This notebook focuses on three main subjects:

1-Test of Normality of the features: We plotted and leveraged the QQ plots of each features to have a better sense of testing the normality of each feature i.e can we assume that each feature distribution among the different neighborhoods have a normal Gaussian distribution or not. You can find the plots from 1-10

2-Visualizing the clusters in the geographical (map) plot of Chicago

You can find the different maps corresponding to clustering result. After applying clustering, we colored the neighborhood in the same cluster with the same color and basically colored the map of Chicago with the corresponding cluster color. It's worth noting that in our clustering implementation, we didn't use any geospatial feature or data regarding the neighborhoods, only redlining, bluelining, canopy,… and we wanted to examine and visualize the effect of clustering when it was applied to a geographical representation. -58

2.5- After the visualization you can see a few chunks which transform the the suicide numbers of each neighborhood to the category that we discussed earlier (i.e 0,1,2,3).

3- Regression/ Prediction Power
After obtaining the clusters, using the clusters label and the features we had already, we tried to evaluate the prediction power of our labels with respect to other features like "Suicide rates" (the very first code chunk starting the prediction power section). For prediction testing, we will examine the prediction power of our initial dataset plus the labels of cluster to predict the suicide rate class.

Following after that, we can see different chunks which test the prediction power against "Housing Energy Usage" , "sales per 100 residential parcels", …

IMPORTANT NOTE!
In our dataset, since our sample size is too small for a machine learning setup (specially testing the prediction power of our clusters) we didn't use the classic cross-fold validation and used the dataset as a whole!
To reproduce the result of linear regressions, testing each of the target variables with respect to the other variables, simply change the target variable to the desired one and fit the regression model created to the data obtained. In our analysis of regression we avoided to do the usual cross-validation technique in our measurement due to small number of samples.