SOM-where in Chicago: What self-organizing maps reveal about ride-sharing in Chicago during the pandemic.

Shakil Rafi

Ph.D. Candidate, Department of Mathematical Sciences
University of Arkansas

OAK Fall 2023

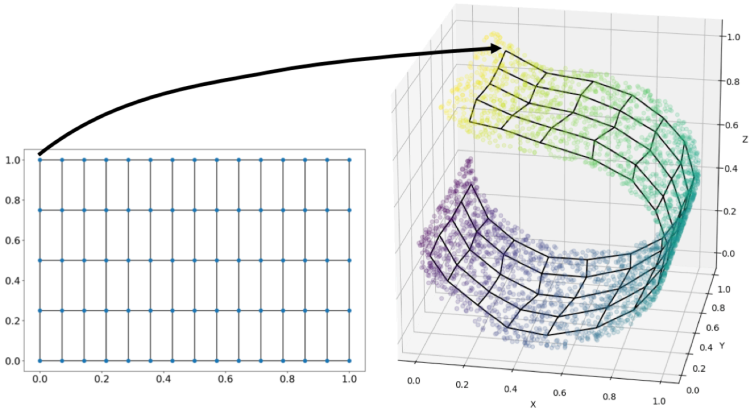
October 26, 2023

Contents

Self-organizing maps, or Kohonen maps

Self-organizing maps are neural networks architectures used primarily for dimensionality reduction.

Proposed by Tuevo Kohonen in 1980s, hence also called Kohonen maps.


1.0
0.8
0.6
0.4
0.2
0.0
0.0
0.2
0.4
0.6
0.8
1.0
X
Y
Z

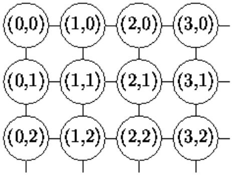
The mathematical description

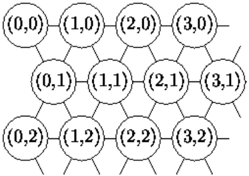
The algorithm

initialize lattice nodes;
initialize weight vectors;
N $\leftarrow$ iteration count;
for i in N:
    $x \leftarrow$ pick random point in dataset;
    $c \leftarrow$ select lattice closest to x;
    move weight vector of c closer to x;
    move weight vectors of the neighbors of c slightly closer to x

Lattice Architecture

Typically we initialize a lattice over the data as either square lattices or hexagonal lattices:





For rectangular lattice Kohonen suggests the (x, y) should be the ratio of the two largest eigenvalues of the autocorrelation matrix.

Weight initialization

The most common ways of initialization is:

Random initialization Slow to converge but because SOMs are fast this may not be an issue

Random sampling initialization Take samples from the dataset.

Training the model: Step 1

Each lattice point has two vectors: w_i representing its weight and l_j representing its position within the lattice.

We start by picking a random point $x \in \mathbb{R}^d$ of our data. We calculate the Euclidean distance of each lattice point from that data and select the lattice point with the smallest such distance:

$$c_i = \arg\min_i \|x - w_i\| \tag{0.1}$$

Where c_i is the index of the best matching unit. BMU.

Training the model: Step 2

We update the weights for all the lattice points for the $k+1$-th iteration:

$$w_i^{(k+1)} = w_i^{(k)} + \alpha_k \cdot \eta_{c_i,i} \cdot \left\| x - w_i^{(k)} \right\| \tag{0.2}$$

Note that α_k is the learning rate s.t.:

$$\alpha_{k+1} \leq \alpha_k \tag{0.3}$$

$$\sum_{k=0}^{\infty} \alpha_k = \infty \tag{0.4}$$

$$\sum_{k=0}^{\infty} \alpha_k^2 \leq \infty \tag{0.5}$$

Note that $\eta_{c_i,i}$ is a neighborhood function, designed to:

Achieve a maximum when $\|l_i - l_j\| = 0$

$\eta_{c_i,i} \to 0$ as $\|l_i - l_j\| \to \infty$

The most common is Gaussian neighborhood:

$$\eta_{c_i,i} = \exp\left[-\frac{\|l_i - l_{c_i}\|}{2\sigma_k^2}\right] \tag{0.6}$$

Error metrics

Most common is quantization error: Let $c = \arg\min_i \|x - w_i\|$. The quantization error is then:

$$E_Q = \sum_i \|x_i - w_c\| \tag{0.7}$$

The topographic error preserves the underlying topology of the data defined as:

$$E_T = \frac{1}{|D|} \sum_{x \in D} te(x) \tag{0.8}$$

where:

$$te(x) = \begin{cases} 1 & c_1 \text{ and } c_2 \text{ are neighbors} \\ 0 & \text{else} \end{cases} \tag{0.9}$$

The dataset

We take inspiration from Soria, Chen, and Stathoupolos, 2019.

We look at ride-share pickups (Uber and Lyft) for the city of Chicago in the year 2020. Our data came from the City of Chicago Data Portal

We collate the data by census tracts, and merge the data with demographics (median income) from the US Census.

And built-environment characteristics (percentage of zero car ownerships, distance to nearest transit) from the EPA smart locations dataset.

The variables

Our variables are as follows:

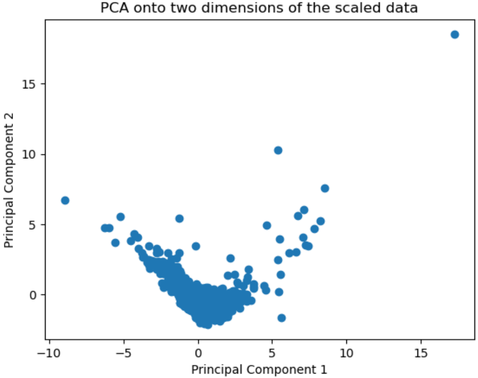
Table 1 Data description

Category	Variables	Unit
Trip Characteristics	Number of Pickups	Number
	Trip Duration	Seconds
	Trip Length	Miles
Socio-demographic Characteristics	Median Income	$
Built Environment	Population Density	People/Acre
	Employment Density	Jobs/Acre
	Land-use Mix	-
	Percentage of zero-car ownership	Percentage
	Distance to nearest transit-stop	Meters

Preliminary analysis

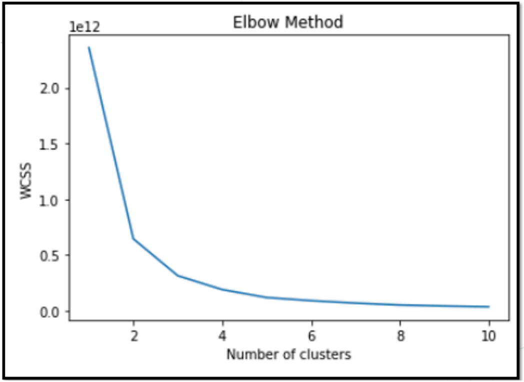
Previous work by the author suggests that much can be gained by doing a segmentation analysis of the dataset.

A principal component analysis of the scaled data shows that there exists an eigenvalue in one direction wit both components accounting for $\approx 50\%$ of the variance



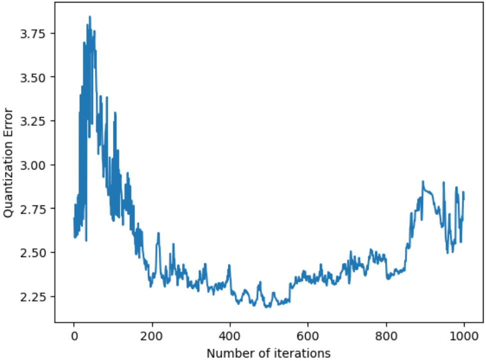
The elbow

Taking a cue from previous work we do create a 2×2 lattice. Four clusters seem to be optimal from an elbow method perspective.



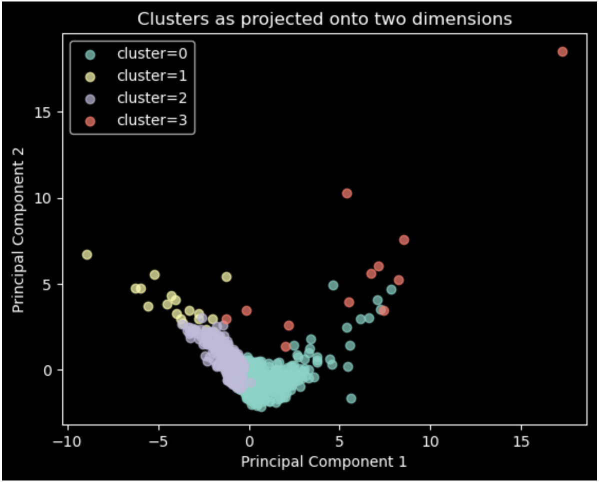
Quantization errors

Quantization errors across iteration counts shows $\sim$ 500 iterations to be optimal



The Clustering

We get a clustering as such:



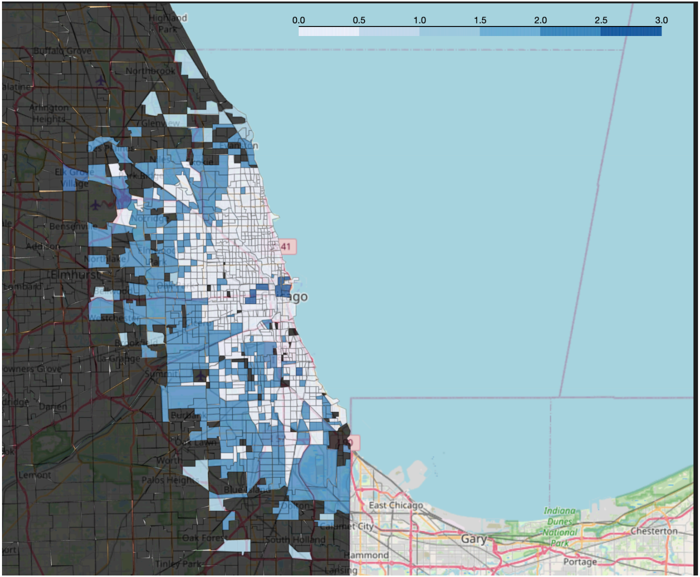
The Clusters I

Cluster	MedIncome	PopDensity	EmpDensity	LUDiversity	% 0 Car
0	$85.9k	38.18	10.2	0.82	24%
1	$149k	10.27	10.46	3.17	10%
2	$73k	18.32	3.41	1.01	14%
3	$164k	31.01	258.6	45.5	19%

The Clusters cont.

Cluster	Median Income	Trip Seconds	Trip Miles
0	$85.9k	1000.3	5.8
1	$149	1961	7.5
2	$73	1249	8.5
3	$164	951	6.14

The map



0.0
0.5
1.0
1.5
2.0
2.5
3.0
Highland Park
Buffalo Grove
Northbrook
Arlington Heights
Glenview
Des Plaines
Evanston
Niles
Skokie
Elk Grove Village
Park Ridge
Norridge
Bensenville
Addison
Northlake
Elmwood Park
Elmhurst
Lombard
Chicago
Westchester
Brookfield
La Grange
Downers Grove
Summit
Darien
Burbank
Oak Lawn
Worth
Lemont
Palos Heights
Blue Island
Dolton
Calumet City
East Chicago
Gary
Indiana Dunes National Park
Chesterton
Portage
Hammond
Oak Forest
South Holland
Tinley Park
Lansing
41

The Clusters cont.

The full table:

cluster	Pickups	MedianIncome	TotalPopulation	Population_Density	Employment_Density	Percent_Zero_Car_Ownership	LandUse_Diversity	Distance_from_transit	Trip_Miles	Trip_Seconds
0	31765.885615	85900.493934	3385.093588	38.185717	10.308518	0.244521	0.824956	233.538478	5.797611	1000.290215
1	297.263158	149221.315789	4601.578947	10.272762	10.465472	0.101122	3.175389	457.476149	17.542535	1961.039958
2	2929.275261	73009.703833	4206.780488	18.320666	3.423521	0.141682	1.018148	362.257844	8.466476	1249.391233
3	406730.166667	164795.250000	4237.083333	31.013539	258.614703	0.192708	45.003452	208.357014	6.146520	951.360888

The takeaways

Key takeaways:

People from richer neighborhoods take shorter Uber trips

Employment density is a better predictor than population density for the number of pickups

Percentage of zero car interacts with median income.

Future work

Possible future work:

Could we do a detailed SHAP analysis of the factors predicting pickups, i.e. is it the case that median income of tracts can be a good predictor?

Could we do a time series analysis and see a seasonality decrease around March 2020.

Bibliography

City of Chicago Data Portal, 2021. Transportation Network Providers - Trips https://data.cityofchicago.org

Ponmalai, Ravi, and Kamath, Chandrika. 2019. "Self-Organizing Maps and Their Applications to Data Analysis". United States. https://doi.org/10.2172/ https://www.osti.gov/servlets/purl/1566795.

R., S.; Nithila, Arna Nishita (2022). Who rides Uber anyway? A census-tract level analysis and clustering of ride-shares for the city of Chicago during the era of the pandemic. TechRxiv. Preprint. https://doi.org/10.36227/techrx

Soria, J., Chen, Y., Stathopoulos,A., 2020. K-Prototypes Segmentation Analysis on Large-Scale Ridesourcing Trip Data, Transportation Research Board 2020, DOI: 10.1177/0361198120929338

Smart Location Database. https://www.epa.gov/smartgrowth/smart- locatior mapping

Check it out: