Machine Learning

Self Organizing Maps (SOM's)

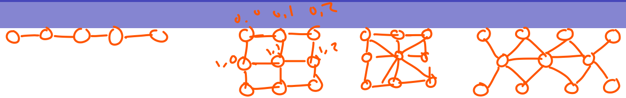
Edgar F. Roman-Rangel.
edgar.roman@itam.mx

Digital Systems Department.
Instituto Tecnológico Autónomo de México, ITAM.

May 22nd, 2021.

Outline

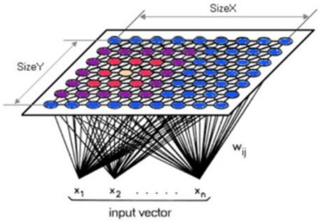
SOM's

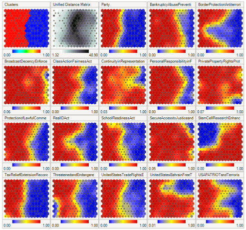


Introduction

Self Organizing Maps (SOM's), a.k.a., Kohonen SOM's.

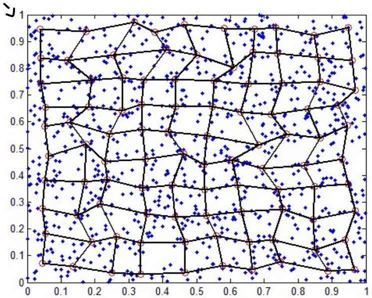
- Competitive learning: data points compete for representation.
- Provides both dimensionality reduction and clustering.
- Use a predefined grid of *centroids*.
- Each data point is represented by both: the index of its closest centroid (cluster), and the coordinates of that cluster (dimensionality reduction).

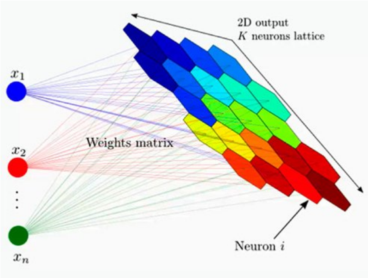




Grid of centroids

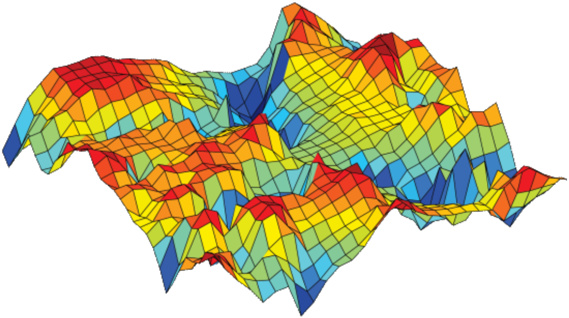
- Each data point influences the position of centroids.
- Use the coordinates of the centroids as weights (parameters).



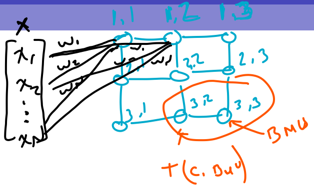


Visualize

Think of them as a thin coating layer, placed nearby a high-D space, where each node in the coating is connected to the points in the high-D space, and points pull down the nodes of this coating.



Process



$$d(x, c) = \sqrt{(x_1 - w_1^{(c)})^2 + (x_2 - w_2^{(c)})^2 + \dots}$$

1. Initialize weights randomly.
2. Randomly select an input vector.
3. Find the best matching unit (BMU).
4. Update unit weights (and neighbors).
5. Repeat from step 2., until done.

Cercanía entre BMU y $c^{(i)}$

Cercanía entre x y $c^{(i)}$

Update weights

For a given point $\mathbf{x}$ and a particular node $n^{(i)}$, do

hyperparametro

$$w_j^{(i)} = w_j^{(i)} + \eta_t T(n^{(i)}) d(\mathbf{x}, n^{(i)}), \qquad \forall j, i,$$

$x \in X$: set de entrenamiento

- w_j: j-th weight of the BMU.
- $\eta_t = \eta_0 e^{(-t/\lambda)}$, $\qquad t = 1, \ldots, T$: number of iterations.
- $\lambda = T/\sigma_0$, $\qquad \sigma_0$: radius of map.
- $T(\cdot) = e^{(-d_{BMU}/2\sigma_t^2)}$: closeness function, w.r.t. BMU.
- $\sigma_t = \sigma_0 e^{(-t/\lambda)}$.
- $d(\cdot)$: Euclidean distance.

Q&A

Thank you!

edgar.roman@itam.mx