MAC 0459 / 5865

Data Science and Engineering

R. Hirata Jr. (hirata@ime.usp.br)

Class 15 (2020)

Some other thoughs and algorithms

Clustering based on graphs

Each item is a vertex of a graph.

Edges conect the vertices according to some criterium (nearest neighbor, similarity, special relation, etc)

Clustering: graph cut, connected components, ...

Clustering based on density

Clusters are regions of high density.

Basic idea: estimate the density of the points in the space and group based on density significance

Good for complex shaped clusters

Outliers are usually handled/discovery efficiently

Have time complexity less than $\mathcal{O}(N^2)$

Clustering to large volumes of data

Data mining applications motivated the creation of a large number of new algorithms mainly to large volumes of data.

Some algorithms:

> CURE, ROCK, Chameleon, k-medoids, Fuzzy, PAM, CLARA, CLARANS, SOM, DBSCAN, DENCLUE, CLIQUE, etc

Clustering to large volumes of data

DBScan - Go to PR slides, page 46

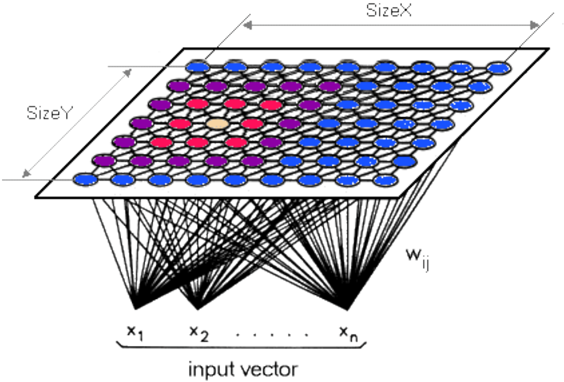
SOM - Self Organized Maps

SOM - Self Organized Maps

Basic idea: map objects in a high dimensional space to a low dimensional space making that objects that are near in high dimension remain near in low dimension.

- **The low dimension space corresponds to a set of nodes organized in a grid in the plane (map)**
- **Each node of the map has a coordinate (in the plane) and a vector of weights of dimension** d
- **OBS.: The literature usually presents as SOM as a kind of neural network**

SOM - Self Organized Maps - Architecture



Orange nodes: BMU (best matching unit), node that has a vector of weights similar to a given input $x \in X$.

Pink and dark blue nodes: neighbors defined by a window function.

SOM - Self Organized Maps - Algorithm

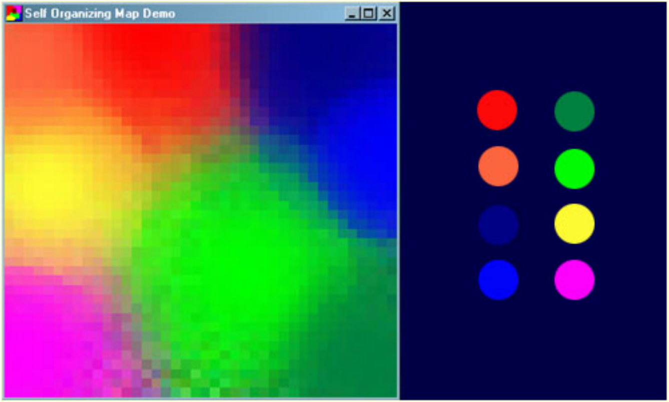
```
Initialize the weight of the nodes of the map
Repeat
    For each $\mathrm{x} \in X$
        Let $\mathrm{p}_k$ be a BMU
        Update the BMU and its neighbors $\mathrm{p}$

            $\mathrm{w}_{ki}(t+1) = \mathrm{w}_{ki}(t) + \eta(t)\phi(\mathrm{p} - \mathrm{p}_k)(\mathrm{x}_i - \mathrm{w}_{ki}(t))$

until convergence
```

ϕ is a window function (kernel function) and $\eta(t)$ is a learning rate.

w_{ki} is the ith component of the weight vector w_k associated to node p_k in the map

SOM - Self Organized Maps

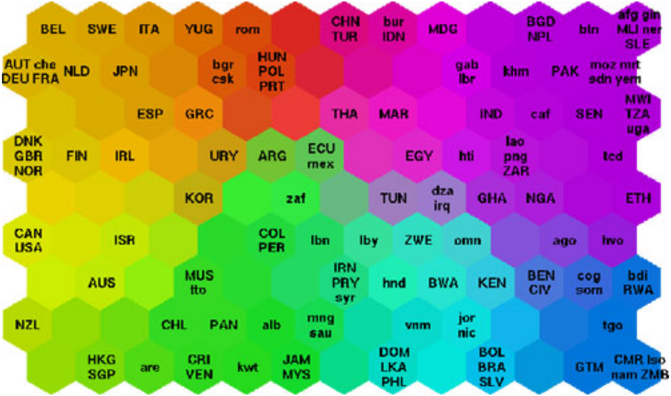
Example: if the weight vector has 3 components, they can be thought as the R, G, B channes and the map can be "painted" by the corresponding RGB color.



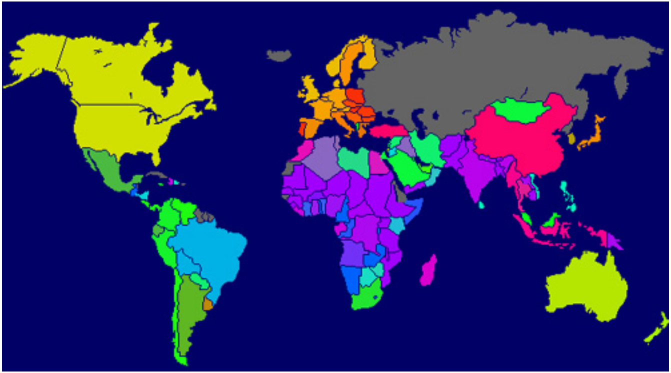
It is not easy to divide the map in regions: how may colors (groups)? To which group the nodes in the border are inside (for instance, between green and blue?)

SOM - Self Organized Maps

Example: The original space are several statistics of a country (education, health, etc)

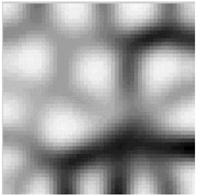


SOM - Self Organized Maps



SOM - Self Organized Maps

Interpretation of a map



Color of the nodes: the intensity represents the difference between nodes (neighbors), for instance, the mean difference between the weight vectors.

Dark lines corresponds to discontinuities and light color regions to similar weight nodes

Each region can be interpreted as a group

We still can apply clustering.

How to validate the clustering result

- Most clustering algorithms **impose** a clustering structure to the dataset
- The dataset may not possess any structure
- **Before** we apply any clustering algorithm, we should check if the dataset has a clustering structure
- This is called **clustering tendency**
- Clustering tendency is heavily based on hypothesis testing

Cluster validation

Cluster validation

- Run an algorithm several times, using different parameters.
- Run different clustering algorithms
- Check with a priori knowledge (area experts)

How do we compare clusters?

Indexes (values between 0 and 1) that indicate how distinct or similar are two partitions:

- Purity
- Rand index
- Mutual information
- etc

Bibliographical references:

- Section 25.1 of Kevin P. Murphy's book (Machine Learning: a Probabilistic Perspective)
- Davies, David L.; Bouldin, Donald W., "A Cluster Separation Measure," Pattern Analysis and Machine Intelligence, IEEE Transactions on , vol.PAMI-1, no.2, pp.224,227, April 1979 (doi: 10.1109/TPAMI.1979.4766909)
- Marina Meila, Comparing clusterings – an information based distance, Journal of Multivariate Analysis, Volume 98, Issue 5, May 2007, Pages 873-895, http://dx.doi.org/10.1016/j.jmva.2006.11.013.

Purity index

N_{ij}: number of itens in cluster i that belong to class j
$N_i = \sum_{j=1}^{c} N_{ij}$: total number of elements in cluster i
$p_{ij} = \frac{N_{ij}}{N_i}$ is the empirical distribution over class labels for cluster i
$p_i = max_j\{p_{ij}\}$ is the purity of a cluster

Overall purity index: $\sum_i \frac{N_i}{N} p_i$

If there is no mixture p_i will be 1 for all i and the index will be 1. On the other hand, the larger the mixture in the clusters i, the lesser will be p_i and so the Purity.

Weak point: it doesn't take in consideration the number of clusters; if all clusters are unitary, they will be pure (the larger the number of clusters, the purer).

Purity index

Example: $U = \big\{\{A, A, A, A, A, B\}, \{A, B, B, B, B, C\}, \{A, A, C, C, C\}\big\}$

Overall purity index: $\sum_i \frac{N_i}{N} p_i = \frac{6}{17}\frac{5}{6} + \frac{6}{17}\frac{4}{6} + \frac{5}{17}\frac{3}{5} = 0.71$

Rand index

Let U and V two different partitions of X.

TP: number of pairs that are in the same cluster in both U and V.

TN: number of pairs that are in different clusters in both U and V.

FN: number of pairs that are in distinct clusters in U but in the same cluster in V

FP: number of pairs that are in the same cluster in U but in different clusters in V

Rand index: $R = \frac{TP+TN}{TP+FP+TN+FN}$

Rand index

Example:

$$\begin{aligned} U &= \{\{A, A, A, A, A, B\}, \{A, B, B, B, B, C\}, \{A, A, C, C, C\}\} \\ V &= \{\{A, A, A, A, A, A, A, A\}, \{B, B, B, B, B\}, \{C, C, C, C\}\} \end{aligned}$$

We have

$TP + FP = C(6, 2) + C(6, 2) + C(5, 2)$

$TP = C(5, 2) + C(4, 2) + C(3, 2) + C(2, 2) = 20$ and

Homework, show that $FN = 24$ e $TN = 72$.

Therefore, the number of possible pairs $(TP + FP + TN + FN)$ is 136 and

$R = \frac{20+72}{136} = 0.68.$