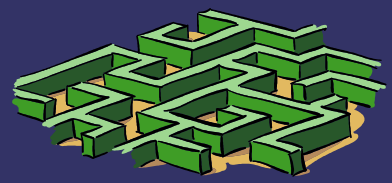
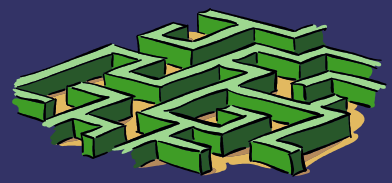
Exploratory Data Analysis using Self-Organizing Maps

Madhumanti Ray



Content

- Introduction
- Data Analysis methods
- Self-Organizing Maps
- Conclusion

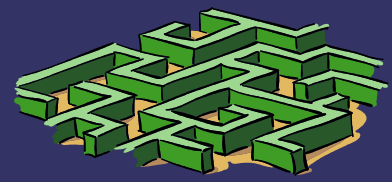


Visualization of high-dimensional data items

- Exploratory data analysis finds structures in a given data set.
- Many methods exist – deal with low dimensional data

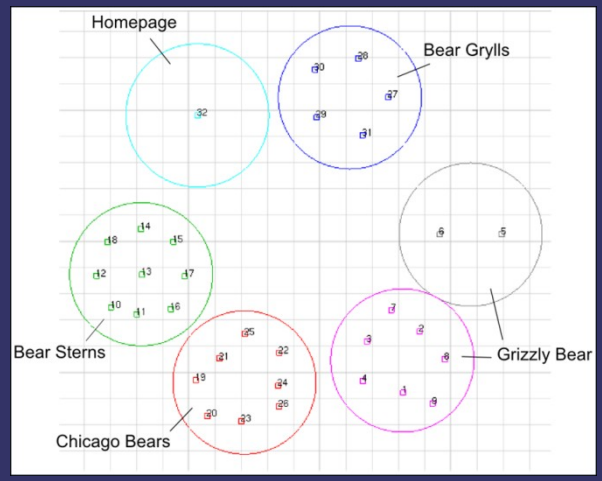
  Eg: Andrews curve, Chernoff' faces

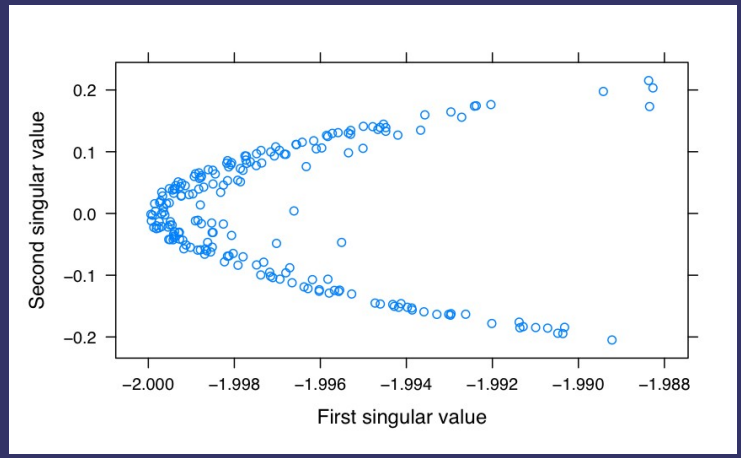
- Drawback? Difficult to visualize high-dimensional data, do not reduce amount of data

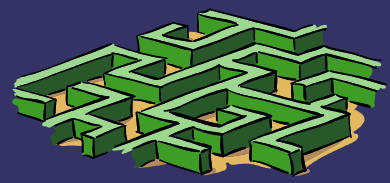


Data Exploration Techniques

- Clustering
- Projection





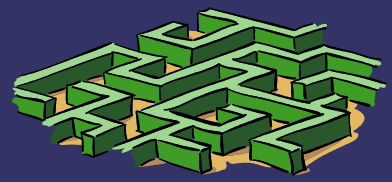


Clustering

Reduces amount of data by grouping similar items together.

- Hierarchical
- Partitioning

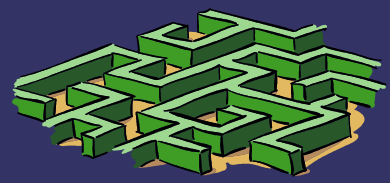
Partitioning : divide the data set into disjoint data clusters



K- means Algorithm

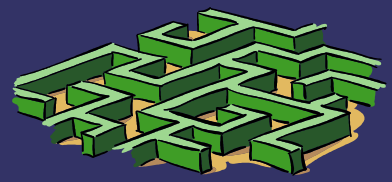
$$\arg\min_{\mathbf{S}} \sum_{i=1}^{k} \sum_{\mathbf{x}_j \in S_i} \|\mathbf{x}_j - \boldsymbol{\mu}_i\|^2$$

- *Find Euclidean distance between each input and centroid of a cluster*
- *Assign input to nearest cluster*
- *Do above steps iteratively till value does not change significantly*



Projection Methods

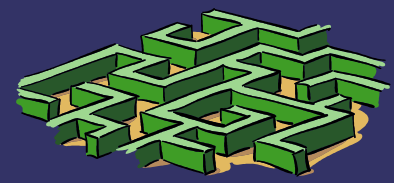
- Represents high-dimensional data in a lower-dimensional space
- Properties of the structure of data set is preserved
    - Linear Projection methods
    - Non-linear projection methods



Linear Projection

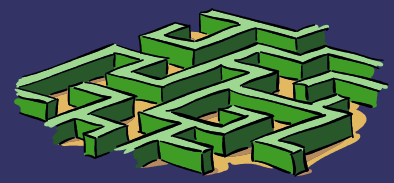
- Principal component analysis:
  - Data sets with linear structures
  - Original variables mapped to few variable, *PC*, retain key features of data set

- Projection Pursuit:
  - Data projection deviating most from normally distribute data is picked for removal



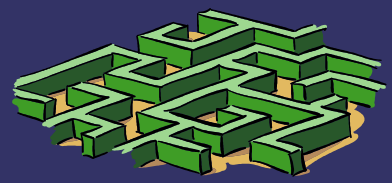
Non-linear Projection methods

- Multidimensional Scaling:
  - Topology preserving
  - Distance between representations in lower-dimensional space close to original distance
  - Objective function: $\Sigma\,[d(k,l) - d'(k,l)]^2$
  - Rank order of distances maintained for comparison maintained by *non-metric MDS* – better measure



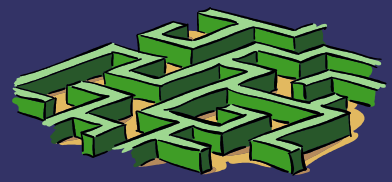
- Sammon's : MDS function normalized by original distance

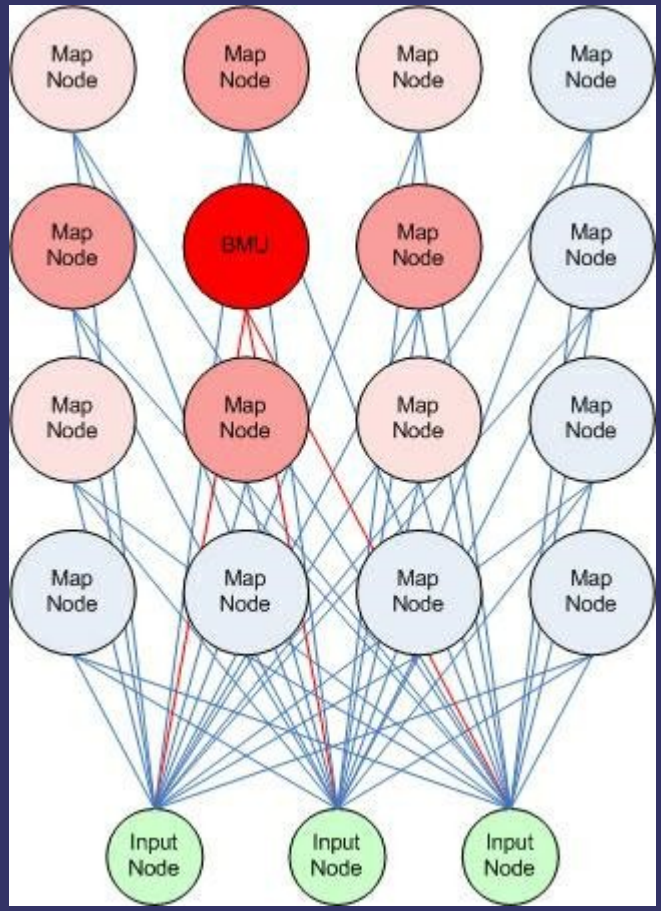
- Principal curves: points on curve is average of all data points projecting to it.

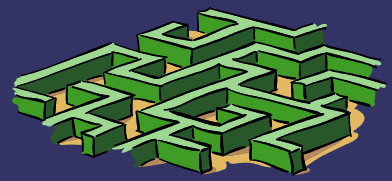


Self Organizing Maps

- Unsupervised learning, non-parametric
- Can do both clustering and projection
- Preserves topology
- Implements competitive learning with cooperation




Map Node
Map Node
Map Node
Map Node
Map Node
BMU
Map Node
Map Node
Map Node
Map Node
Map Node
Map Node
Map Node
Map Node
Map Node
Map Node
Input Node
Input Node
Input Node

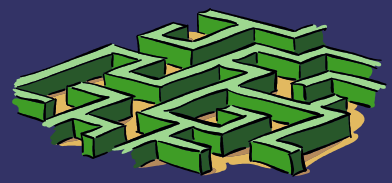


Self Organizing Maps

- Winning unit on lattice has reference vector closest to input:

$$c = c(x) = arg\ min\{||x-m_i||^2\}$$

euclidean distance to find winning unit

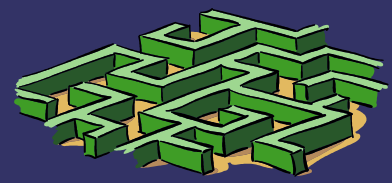


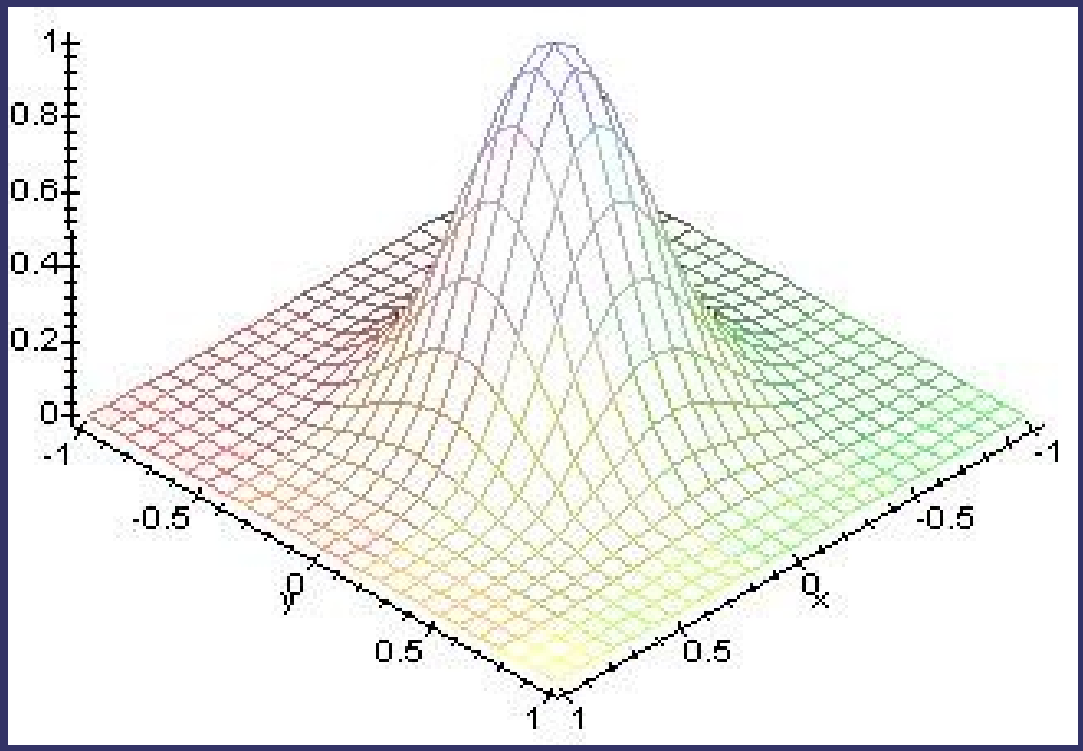
Self Organizing Maps

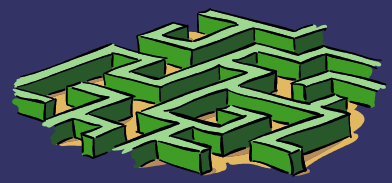
- The learning is shared by neighbouring units - they orient their vectors towards input
- Governed by neighbourhood kernel h – decreasing function

$$m_i(t+1) = m_i(t) + h_{ci}(t)[x(t) - m_i(t)]$$

h is wide at first, reduces over learning

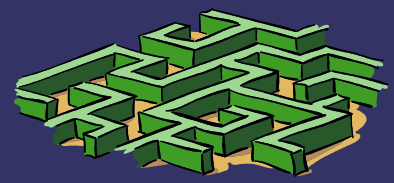



1
0.8
0.6
0.4
0.2
0
-1
-0.5
0
0.5
1
y
1
0.5
0
-0.5
-1
x



Features of SOM useful in exploring data

- *Ordered display*: neighbouring units on the map will represent inputs that are similar
- *Visualization of clusters*: distance between reference vectors gives density of cluster – used for gray level values.
- *Missing data* : reference vectors compared to input using features present in input
- *Outliers*: affect only one unit and neighbours.



Using SOM

Preprocessing:

Feature extraction that represent the variations of data best.

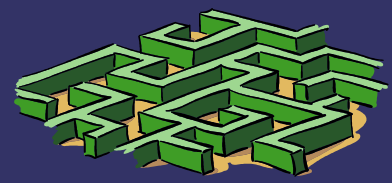
Ordering/scaling among feature components.

Computation of maps:

Initialize reference vectors, choose wide *h*

Neighbourhood kernel *h* reduces in phase one

Phase two, reaches final state of map



Using SOM

Choosing best map:

Teach set of maps with each size of maps and *h*

Pick map that minimizes cost function best from each set

Apply goodness measure for final pick

Interpretation of map:

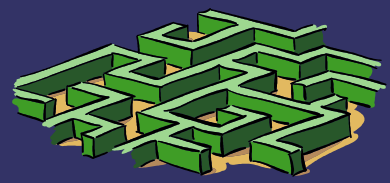
Subjective to data set and problem

SOM maps focused on local relations of data items

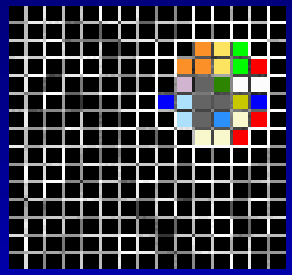
Evaluation:

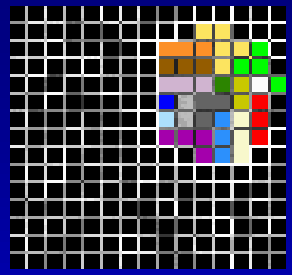
Many methods, eg. Classification accuracy

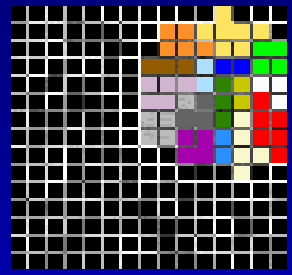
Generalizability

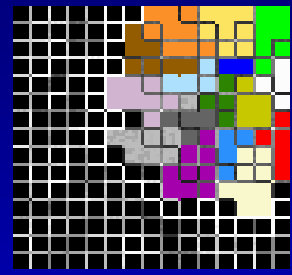


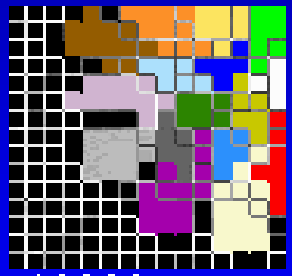
Clustering with SOMs

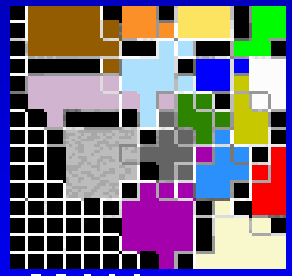

200 steps

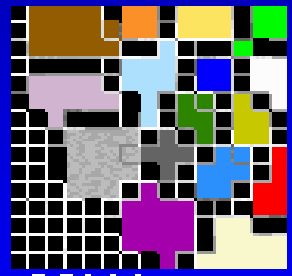

500 steps

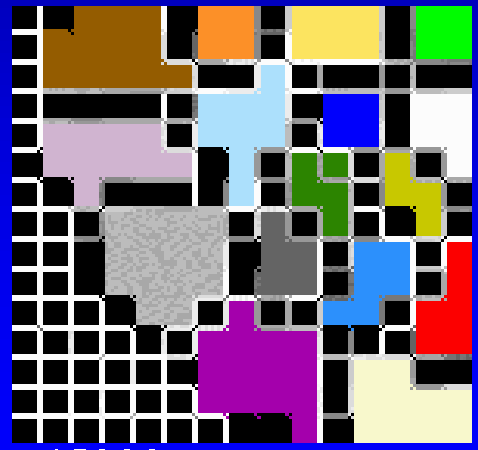

1000 steps

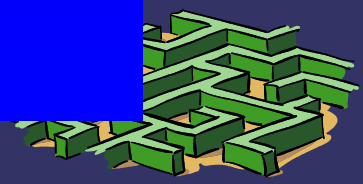

3000 steps


10000 steps


22000 steps


35000 steps

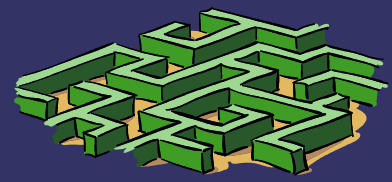

45000 steps



Difference with other MDS

With MDS:

- Topology preserving, not distance i.e. similar inputs mapped to nearby locations
- Computational complexity depends on map units *k*. Each learning step is *O(k)* computations
- No local minima. Start with wide *h*, shrink to narrow form



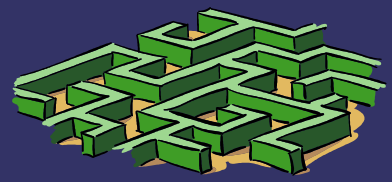
Conclusion

SOM:

Viable alternative to other data exploration methods

One method to implement clustering and projection

Robust to handle continuous, discrete values, sparse data etc.



Questions?