Knowledge Discovery and Data Mining

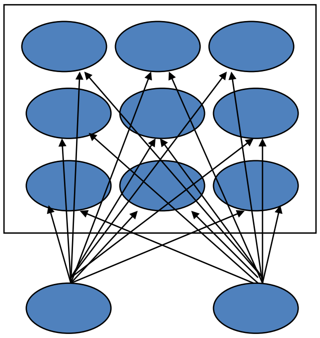
Unit # 14

Kohonen Map

- Kohonen networks are a type of neural network that perform clustering, also known as a knet or a self-organizing map.
- This type of network can be used to cluster the data set into distinct groups when you don't know what those groups are at the beginning.
- Records are grouped so that records within a group or cluster tend to be similar to each other, and records in different groups are dissimilar.
- The basic units are neurons, and they are organized into two layers: the input layer and the output layer (also called the output map).

Kohonen Map (Cont'd)

- Formalized by Teuvo Kohonen in 1982 for unsupervised clustering.
- All of the input neurons are connected to all of the output neurons, and these connections have strengths, or weights, associated with them.
- During training, each unit competes with all of the others to "win" each record.
- Input data is presented to the input layer, and the values are propagated to the output layer. The output neuron with the strongest response is said to be the winner and is the answer for that input.



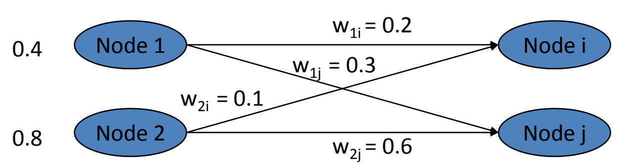
Kohonen Map (Cont'd)

- Initially, all weights are random. When a unit wins a record, its weights (along with those of other nearby units, collectively referred to as a neighborhood) are adjusted to better match the pattern of predictor values for that record.
- All of the input records are shown, and weights are updated accordingly. This process is repeated many times until the changes become very small.
- As training proceeds, the weights on the grid units are adjusted so that they form a two-dimensional "map" of the clusters (hence the term self-organizing map).

Kohonen Map (Cont'd)

- When the network is fully trained, records that are similar should appear close together on the output map, whereas records that are vastly different will appear far apart.
- Usually, a Kohonen net will end up with a few units that summarize many observations (strong units), and several units that don't really correspond to any of the observations (weak units). The strong units (and sometimes other units adjacent to them in the grid) represent probable cluster centers.

Working of Kohonen Maps



- The score for classifying a new instance with output node j is given by

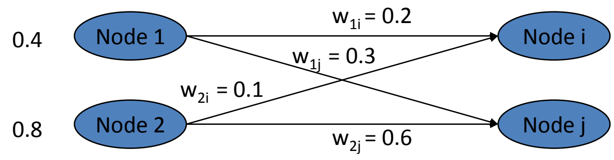
$$\text{sqrt}\,(\sum (n_i - w_{ij})^2)$$

  - n_i is the attribute value for the current instance at input i.
  - w_{ij} is the weight associated with the ith input node and output node j.
- Weights are updated according the following formula:

$$w_{ij}\text{ (new)} = w_{ij}\text{ (current)} + \Delta w_{ij}$$

  - where $\Delta w_{ij} = r(n_i - w_{ij})$, r is the learning parameter and $0<r<1$.

Working of Kohonen Maps (Cont’d)



- Score of Node i: sqrt(0.4-0.2)^2 + (0.8-0.1)^2 = 0.53
- Score of Node j: sqrt(0.4-0.3)^2 + (0.8-0.6)^2 = 0.05

- Thus, the record belongs to Cluster j.

- Next we update the weights of incoming links to node j. Let r = 0.8
- $\Delta w_{1j} = 0.8 \times (0.4 – 0.3) = 0.08$
- $\Delta w_{2j} = 0.8 \times (0.8 – 0.6) = 0.16$
- $w_{1j} = 0.3 + 0.08 = 0.38$
- $w_{2j} = 0.6 + 0.16 = 0.78$

Kohonen Map (Cont’d)

- The simplicity of this algorithm makes it a great choice for clustering.
- One primary disadvantage of the algorithm is that the number of output classes must be defined upfront.
- This is significant because it assumes that we have some general knowledge of the data and how it should be classified.

Adaptive Resonance Theory

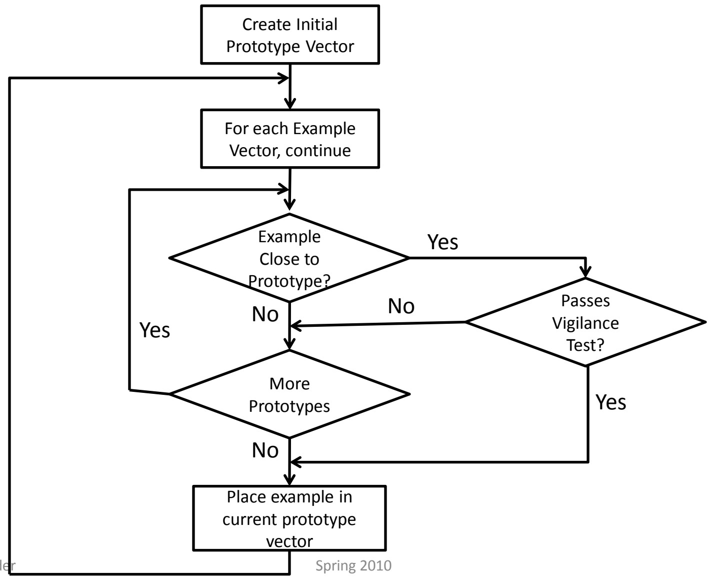
- The ART1 (adaptive resonance theory) algorithm is a simple, unsupervised learning algorithm with biological motivation.
- It works with objects called feature vectors.
- A feature vector is nothing more than a collection of binary values that represent some type of information.
- An example of a feature vector is a customer's purchase data

Hammer	Paper	Pen	Kit-Kat	Pencil	Binder	Snickers
1	0	0	1	0	0	1

Working of ART1

- We being with a set of feature vectors and a set of initialized prototype vectors (P1….PN).
- The prototype vector is the center of the cluster.
- The number of prototype vector, N, is the maximum number of clusters that can be supported.
- We initialize a vigilance parameter (ρ) to a small value between 0 and 1 and a beta parameter to a small positive integer.
- The parameter (d) represents the dimension of the vectors.

ART1 Algorithm Flow



ART1 Conditions

- Initially, no prototype vectors exist, so at the start fo the algorithm an initial prototype vector is created with the first example vector.
- We then check all subsequent example feature vectors against each existing prototype vector for its proximity.
- Proximity Test
  - $||P_i \cap E|| / (\beta + ||Pi||) > ||E|| / (\beta + d)$
- Vigilance Test
  - $||P_i \cap E|| / (||E||) > \rho$

Example

- P0: {1, 0, 0, 1, 1, 0, 1}
- P1: {1, 1, 0, 0, 0, 1, 0}
- E: {1, 1, 1, 0, 0, 1, 0}
- $\beta = 1.0$, $\rho = 0.6$, $d = 7$
- Proximity Test with P0 ?
- Proximity Test with P1 ?
- Vigilance Test with P1 ?
- P1 AND E {1,1,0,0,0,1,0} AND {1,1,1,0,0,1,0} = {1,1,0,0,0,1,0}

Example (Cont'd)

- Proximity Test
  - $||P_i \cap E|| / (\beta + ||Pi||) > ||E|| / (\beta + d)$
- Vigilance Test
  - $||P_i \cap E|| / (||E||) > \rho$
- P & E = 3, P = 3, E = 4, beta + d = 8
- 3 / 4 > 4 / 8
- 3 / 4 > 0.6

Euclidean ART

- Let
  - Cluster Set = CS= Empty Set
  - Dataset = D = Total customer accounts
  - Data row =R = A single data record
  - Vigilance= v = parameter to control distance between data points in cluster
  - n= Total Number of iterations specified by the user

Euclidean ART (Cont'd)

- While (Iterations < n)
  - Begin
    - Create a single cluster and add it to CS
    - Initialize its centroid to the first record
    - Foreach record R in dataset D
      - Compute Euclidean distance between R and centroid of all the clusters in CS
      - Find the minimum distance and denote it as mindist
      - If the (mindist < v)
        - » Add R into the cluster c and recompute the centroid of c
        - » Increment cluster size of c
      - Else
        - » Create a new cluster and add this new cluster to CS
        - » Set R as the initial centroid of the new cluster
    - End // foreach
    - Reshuffle the items in data set D so as to remove any bias and repeat the above steps
  - End // total iterations

Termination Condition

- As example feature vectors are tested against the prototype vectors, new clusters are created or existing clusters are modified at the inclusion of an example.
- This process, known as “resonance”, indicates the process of learning within the algorithm.
- When the algorithm reaches equilibrium (that is, no further changes occur with the prototype vectors), learning is complete and the data set is classified.

ART Summary

- ART1 is both conceptually simple and easy to implement.
- Earlier algorithms, such a k-means clustering algorithm, though much simpler, have some significant drawbacks.
- For example, k-means does not allow the creation of new clusters (the clusters are statically defined at the start).
- Also, no parameter exists within k-means to adjust the class size of the result clusters.
- A drawback to both algorithms (ART1 and k-means) is that the final set of clusters (and prototype vectors) can be influenced based on the order in which training is performed.