Data Mining

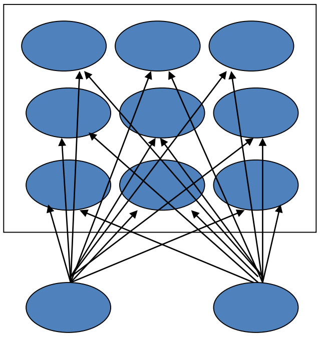
Unit # 9

Kohonen Map

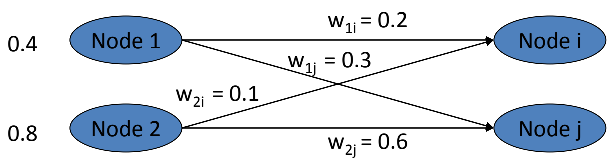
- Kohonen networks are a type of neural network that perform clustering, also known as a knet or a self-organizing map.
- This type of network can be used to cluster the data set into distinct groups when you don't know what those groups are at the beginning.
- Records are grouped so that records within 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.



Working of Kohonen Maps



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

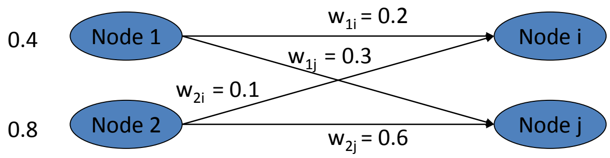
  $$\text{sqrt}\left(\sum (n_i - w_{ij})^2\right)$$

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

- Thus, the record belongs to Cluster j.

- Next we update the weights of incoming links to node j. Let r = 0.8
- $\Delta w_{1j} = 0.8 \times (0.4 – 0.3) = 0.08$
- $\Delta w_{2j} = 0.8 \times (0.8 – 0.6) = 0.16$
- $w_{1j} = 0.3 + 0.08 = 0.38$
- $w_{2j} = 0.6 + 0.16 = 0.76$

Summary

- 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).

Summary (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.

RECAP OF FEW CONCEPTS

Bagging vs. Boosting

- Suppose we run five different models on a given classification problem, and got the following accuracy on the test data set.
  - Model M1: 0.8
  - Model M2: 0.7
  - Model M3: 0.95
  - Model M4: 0.52
  - Model M5: 0.6
- A new record has arrived and we pass it to all 5 models and their predictions are as follows:
  - M1: True, M2: False, M3: False, M4: False, M5: True

Bagging vs. Boosting (Cont’d)

- What should be the final prediction of Bagging and Boosting?
- For Bagging, we simply do the majority voting. 3 models are classifying the record as “False” while the other two are classifying as “True”. Hence Bagging would declare it as “False”.
- In Boosting, we first compute the weight of each model: log[(1-error(Mi)/error(Mi)]

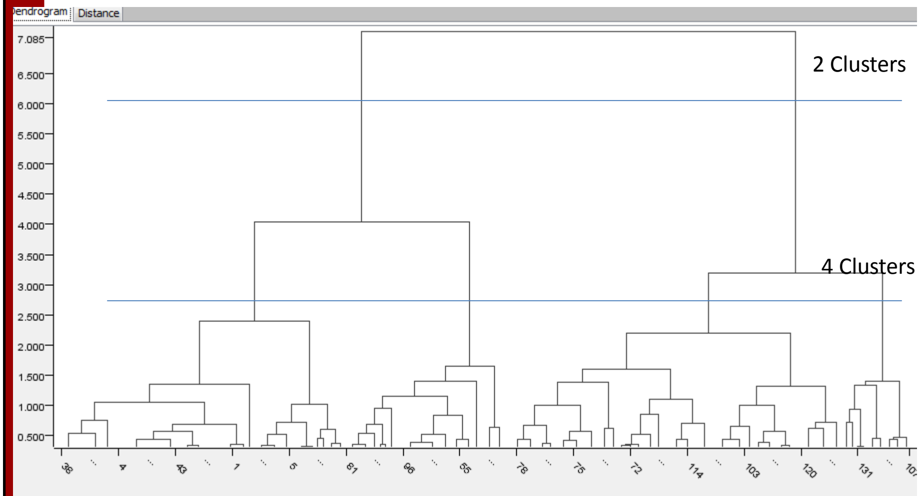
Bagging vs. Boosting (Cont'd)

- 3 models (M2, M3, M4) are favoring "False" and their weights are log(0.7/0.3), log(0.95/0.05), log(0.52/0.48), respectively.
- 2 models (M1, M4) are favoring "True" and their weights are log(0.8/0.2) and log(0.6/0.4), respectively.
- Weight of False is 1.68 while weight of True is 0.778.
- Hence, the prediction by boosting is "False".

Bootstrap Example

- Suppose we have following elements:
  - 1, 14, 8, 9, 6, 2, 5, 7, 11, 15, 3, 18
- If we do a simple random sample (without replacement) of 5 elements then our sample **won't** have any repeated entries.
  - 14, 2, 7, 8, 3
- But if we do with replacement sampling of 5 elements then our sample **may** have repeated entries:
  - **8**, 11, 3, **8**, 6 (8 is repeated twice)

Guessing Number of Clusters from Denogram



Clustering - Handson

- Develop dendrogram of iris and segment-challenge data set.
- Interpret the centroids obtained via k-Means.
- Compare the performance of k-Means and Fuzzy c-Means via entropy scorer node in KNIME.