Unsupervised Learning

(chapter 11)

Kai Goebel, Bill Cheetham

GE Corporate Research & Development

goebel@cs.rpi.edu

cheetham@cs.rpi.edu

Stuff we will talk about

Competitive Learning Networks

Kohonen Self-Organizing Networks

Learning Vector Quantization

Introduction

When no teacher or critic is available, only the input vectors can be used for learning.

Learning system categorizes or detects features without feedback

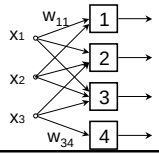
=> use for clustering, feature extraction, similarity detection, data mining

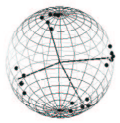
Competitive Learning

Winner takes all:

Weights of the unit with highest activation are updated

The weight vector rotates slowly towards the cluster centers





Competitive Learning

Activation of output unit j

$$a_j = \sqrt{\sum_{i=1}^{3} \left(x_i - w_{ij}\right)^2} = \left\| x - w_j \right\|$$

and the weights are updated according to:

$$w_k(t+1) = w_k(t) + \eta\left(x(t) - w_k(t)\right)$$

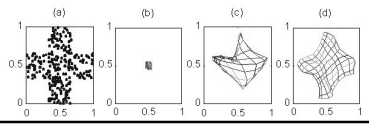
Note: different metrics can be used leading to different solutions

If initial weights are far from actual centers: may never get updated;

=> use leaky learning

Self-Organizing Networks (Kohonen)

- Learning is similar to Competitive Learning
- Not only the winning units are updated but all the weights in a neighborhood
- Size of neighborhood decreases
- Topological properties in the input data is reflected in the output units through neighborhood constraints



Self-Organizing Networks

Algorithm

- Step 1

Select winning output (smallest dissimilarity)

$$\|x - w_c\| = \min_i \|x - w_i\|$$

- Step 2

Update winners and neighborhood NB_c of winner c

$$\Delta w_i = \eta(x - w_i) \qquad i \in NB_c$$

Reduce η gradually

Learning Vector Quantization (LVQ)

Adaptive data classification

1. Cluster data (using any clustering tool)

2. Label data using majority of data in cluster "voting method"

Fine tune class information to minimize errors by

3. Finding cluster center closest to input

4. If x and w_k belong to the same class:

$$\Delta w_k = \eta(x - w_k)$$

otherwise

$$\Delta w_k = -\eta(x - w_k)$$

Repeat until max. number of iterations reached

Adaptive Resonance Theory (ART)

- Accept and adapt stored prototypes of a class when the input is sufficiently similar to it.

- Input and stored prototype are said to "resonate".

- If the input is not sufficiently similar to any class, then form a new class.

- Similarity is judged by a "vigilance" level

- ART1 focuses on binary input.

- ART2 designed for continuous input.

ART1

Algorithm

1. Enable all output units

2. Find the winner among all enabled output units by comparing the components of the input vector.

3. Check whether the match is good enough by comparing the ratio of bits in input and prototype to vigilance level

4. If the match is good, adjust winning vector by adjusting any bits that are not also in input vector.

5. If the match is not good enough, create a new class by adding the current vector as a class prototype.

last slide