Using Hamming Net in a hand-written text recognition

Oleg Petšjonkin
Introduction

- **Hamming Net**
  - Finds Hamming Distance between input pattern and each sample

- **MAX Net**
  - Uses Hamming Net output as input
  - Finds node with highest output

- Example
Hamming Distance (1)

- Hamming Distance between two binary vectors \( A \) and \( B \) of the same length is the number of places in which \( a_i \) and \( b_j \) differ.

\[
\begin{align*}
A &= (1, 1, -1, 1, 1, -1, -1) \\
B &= (1, -1, 1, 1, -1, -1, 1)
\end{align*}
\]

\[HD(A, B) = 4\]
Hamming Distance(2)

- Geometrical Interpretation
Hamming Distance(3)

\[ A^t B = \left[ n - \underbrace{\text{HD}(A, B)}_{\text{bits are agreed}} \right] - \underbrace{\text{HD}(A, B)}_{\text{bits are different}} \]

\[ = n - 2\text{HD}(A, B) \]
Hamming Net

\[ w_{jk} = \frac{x_j^k}{2}, \]
\[ j = 0 \ldots n - 1, \]
\[ k = 0 \ldots m - 1, \]
\[ T_k = \frac{n}{2} \]
\[ y_k = \sum_{i=0}^{n-1} w_{ik} x_i + T_k \]
Hamming Net(2)

- Activation function

\[ f(\text{net}) = \frac{\text{net}}{n} \]
MAX Net

- Purpose: let $\max(y_1, y_2, y_3, \ldots, y_m) = 1$, and let others $= 0$.
- Activation function:

$$f(net) = \begin{cases} 
0, & net < 0 \\ 
net, & net \geq 0 
\end{cases}$$
MAX Net(2)

\[ y_{i}^{k+1} = f \left( y_{i}^{k} - \sum_{j} (\varepsilon \cdot y_{j}^{k}) \right) \]

where \( i=1 \ldots m; \ j=1 \ldots m; \ j \neq i \)
Hamming Net

- Hamming Net only tells only sample number, not pattern.
Questions?
Demo
Thank You