Lecture Slides for

INTRODUCTION TO

Machine Learning

ETHEM ALPAYDIN
© The MIT Press, 2004

alpaydin@boun.edu.tr
http://www.cmpe.boun.edu.tr/~ethem/i2ml
CHAPTER 12:
Local Models
Introduction

- Divide the input space into local regions and learn simple (constant/linear) models in each patch

- Unsupervised: Competitive, online clustering
- Supervised: Radial-basis func, mixture of experts
Competitive Learning

\[ E(\{m_i\}_{i=1}^{k}|x) = \frac{1}{2} \sum_t \sum_i b_i^t \|x^t - m_i\|^2 \]

\[ b_i^t = \begin{cases} 
1 \quad \text{if } \|x^t - m_i\| = \min_l \|x^t - m_l\| \\
0 \quad \text{otherwise}
\end{cases} \]

- Batch k-means: \[ m_i = \frac{\sum_t b_i^t x^t}{\sum_t b_i^t} \]

- Online k-means:

\[ \Delta m_{ij} = -\eta \frac{\partial E^t}{\partial m_{ij}} = \eta b_i^t (x_j^t - m_{ij}) \]
Initialize $m_i, i = 1, \ldots, k$, for example, to $k$ random $x^t$

Repeat

For all $x^t \in \mathcal{X}$ in random order

$$i \leftarrow \text{arg min}_j \| x^t - m_j \|$$

$$m_i \leftarrow m_i + \eta (x^t - m_j)$$

Until $m_i$ converge

**Winner-take-all network**
Adaptive Resonance Theory

- Incremental; add a new cluster if not covered; defined by vigilance, $\rho$

$$ b_i = \| m_i - x^t \| = \min_{l=1}^{k} \| m_l - x^t \| $$

$$ \begin{cases} 
  m_{k+1} \leftarrow x^t & \text{if } b_i > \rho \\
  \Delta m_i = \eta (x - m_i) & \text{otherwise}
\end{cases} $$

(Carpenter and Grossberg, 1988)
Self-Organizing Maps

- Units have a neighborhood defined; \( m_i \) is “between” \( m_{i-1} \) and \( m_{i+1} \), and are all updated together.
- One-dim map: \( \Delta m_l = \eta \ e(l, i) (x^t - m_l) \)

(Kohonen, 1990)

\[
e(l, i) = \frac{1}{\sqrt{2\pi \sigma}} \exp \left[ -\frac{(l - i)^2}{2\sigma^2} \right]
\]
Radial-Basis Functions

- Locally-tuned units:

\[ p_h^t = \exp \left[ -\frac{\| x^t - m_h \|^2}{2s_h^2} \right] \]

\[ y^t = \sum_{h=1}^{H} w_h p_h^t + w_0 \]
Local vs Distributed Representation

Local representation in the space of \((p_1, p_2, p_3)\)
- \(x^a: (1.0, 0.0, 0.0)\)
- \(x^b: (0.0, 0.0, 1.0)\)
- \(x^c: (1.0, 1.0, 0.0)\)

Distributed representation in the space of \((h_1, h_2)\)
- \(x^a: (1.0, 1.0)\)
- \(x^b: (0.0, 1.0)\)
- \(x^c: (1.0, 0.0)\)
Training RBF

- Hybrid learning:
  - First layer centers and spreads:
    Unsupervised $k$-means
  - Second layer weights:
    Supervised gradient-descent
- Fully supervised
- (Broomhead and Lowe, 1988; Moody and Darken, 1989)
Regression

\[ E(\{m_h, s_h, w_{ih}\}_{i,h}|X) = \frac{1}{2} \sum_t \sum_i (r_i^t - y_i^t)^2 \]

\[ y_i^t = \sum_{h=1}^{H} w_{ih} p_h^t + w_{i0} \]

\[ \Delta w_{ih} = \eta \sum_t (r_i^t - y_i^t) p_h^t \]

\[ \Delta m_{h,j} = \eta \sum_t \left[ \sum_i (r_i^t - y_i^t) w_{ih} \right] p_h^t \frac{(x_j^t - m_{h,j})}{s_h^2} \]

\[ \Delta s_h = \eta \sum_t \left[ \sum_i (r_i^t - y_i^t) w_{ih} \right] p_h^t \|x^t - m_h\|^2 \frac{1}{s_h^3} \]
Classification

\[
E(\{m_{ih}, s_h, w_{ih}\}_{i,h} | X) = - \sum_t \sum_i r_i^t \log y_i^t
\]

\[
y_i^t = \frac{\exp \left[ \sum_h w_{ih} p_h^t + w_{i0} \right]}{\sum_k \exp \left[ \sum_h w_{kh} p_h^t + w_{k0} \right]}
\]
Rules and Exceptions

\[ y^t = \sum_{h=1}^{H} w_h p^t_h + \mathbf{v}^T \mathbf{x}^t + \mathbf{v}_0 \]

Exceptions

Default rule
Rule-Based Knowledge

\[
\text{IF } ((x_1 \approx a) \text{ AND } (x_2 \approx b)) \text{ OR } (x_3 \approx c) \text{ THEN } y = 0.1
\]

\[
p_1 = \exp \left[ -\frac{(x_1 - a)^2}{2s_1^2} \right] \cdot \exp \left[ -\frac{(x_2 - b)^2}{2s_2^2} \right] \text{ with } w_1 = 0.1
\]

\[
p_2 = \exp \left[ -\frac{(x_3 - c)^2}{2s_3^2} \right] \text{ with } w_2 = 0.1
\]

- Incorporation of prior knowledge (before training)
- Rule extraction (after training) (Tresp et al., 1997)
- Fuzzy membership functions and fuzzy rules
Normalized Basis Functions

\[
g_h^t = \frac{p_h^t}{\sum_{l=1}^{H} p_l^t} = \frac{\exp[-\|x^t - m_h\|^2 / 2s_h^2]}{\sum_l \exp[-\|x^t - m_l\|^2 / 2s_l^2]}
\]

\[
y_i^t = \sum_{h=1}^{H} w_{ih} g_h^t
\]

\[
\Delta w_{ih} = \eta \sum_t (r_i^t - y_i^t) g_h^t
\]

\[
\Delta m_{hj} = \eta \sum_t \sum_i (r_i^t - y_i^t) (w_{ih} - y_i^t) g_h^t \frac{(x_j^t - m_{hj})}{s_h^2}
\]
Competitive Basis Functions

- Mixture model:  
  \[ p(r^t | x^t) = \sum_{h=1}^{H} p(h | x^t) p(r^t | h, x^t) \]

\[
p(h | x) = \frac{p(x | h) p(h)}{\sum_l p(x | l) p(l)}
\]

\[
g^t_h = \frac{a_h \exp[-\|x^t - m_h\|^2 / 2s_h^2]}{\sum_l a_l \exp[-\|x^t - m_l\|^2 / 2s_l^2]}
\]
\textbf{Regression} 

\[ p(r^t | x^t) = \prod_i \frac{1}{\sqrt{2\pi}\sigma} \exp \left[ -\frac{(r^t_i - y^t_i)^2}{2\sigma^2} \right] \]

\[ \mathcal{L}(\{m_h, s_h, w_{ih}\}_{i, h} | X) = \sum_t \log \sum_h g_h^t \exp \left[ -\frac{1}{2} \sum_i (r^t_i - y^t_{ih})^2 \right] \]

\[ y^t_{ih} = w_{ih} \text{ is the constant fit} \]

\[ \Delta w_{ih} = \eta \sum_t (r^t_i - y^t_{ih}) f_h^t \quad \Delta m_{h,j} = \eta \sum_t (f_h^t - g_h^t) \frac{(x^t_j - m_{h,j})}{s_h^2} \]

\[ f_h^t = \frac{g_h^t \exp\left[ -\frac{1}{2} \sum_i (r^t_i - y^t_{ih})^2 \right]}{\sum_l g_l^t \exp\left[ -\frac{1}{2} \sum_i (r^t_i - y^t_{il})^2 \right]} \]

\[ p(h | r, x) = \frac{p(h | x)p(r | h, x)}{\sum_l p(l | x)p(r | l, x)} \]
Classification

\[ \mathcal{L}(\{m_h, s_h, w_{ih}\}_{i,h} | X) = \sum_t \log \sum_h g^t_h \prod_i (y_{ih}^t)^{r_i^t} = \sum_t \log \sum_h g^t_h \exp \left[ \sum_i r_i^t \log y_{ih}^t \right] \]

\[ y_{ih}^t = \frac{\exp w_{ih}}{\sum_k \exp w_{kh}} \]

\[ f_h^t = \frac{g_h^t \exp \left[ \sum_i r_i^t \log y_{ih}^t \right]}{\sum_l g_i^t \exp \left[ \sum_i r_i^t \log y_{il}^t \right]} \]
EM for RBF (Supervised EM)

- E-step: \( f_h^t \equiv p(r | h, x^t) \)

- M-step:
  \[
  m_h = \frac{\sum_t f_h^t x^t}{\sum_t f_h^t}
  \]
  \[
  s_h = \frac{\sum_t f_h^t (x^t - m_h)(x^t - m_h)^T}{\sum_t f_h^t}
  \]
  \[
  w_{ih} = \frac{\sum_t f_h^t r_{i}^t}{\sum_t f_h^t}
  \]
Learning Vector Quantization

- $H$ units per class prelabeled (Kohonen, 1990)
- Given $x$, $m_i$ is the closest:

$$\begin{cases} 
\Delta m_i = \eta (x^t - m_i) & \text{if } x^t \text{ and } m_i \text{ have the same class label} \\
\Delta m_i = -\eta (x^t - m_i) & \text{otherwise}
\end{cases}$$
Mixture of Experts

- In RBF, each local fit is a constant, $w_{ih}$, second layer weight
- In MoE, each local fit is a linear function of $x$, a local expert:

$$w_{ih}^t = v_{ih}^T x^t$$

(Jacobs et al., 1991)
MoE as Models Combined

- Radial gating:

\[ g_h^t = \frac{\exp[-\|x^t - m_h\|^2/2s_h^2]}{\sum_l \exp[-\|x^t - m_l\|^2/2s_l^2]} \]

- Softmax gating:

\[ g_h^t = \frac{\exp[m_h^T x^t]}{\sum_l \exp[m_l^T x^t]} \]
Cooperative MoE

- Regression

\[
E(\{m_h, s_h, w_{ih}\}_{i,h} | X) = \frac{1}{2} \sum_t \sum_i (r_i^t - y_i^t)^2
\]

\[
\Delta \nu_{ih} = \eta \sum_t (r_i^t - y_i^t) g_h^t x_t
\]

\[
\Delta m_{h,j} = \eta \sum_t \sum_i (r_i^t - y_i^t) (w_{ih}^t - y_i^t) g_h^t x_j^t
\]
Competitive MoE: Regression

\[ \mathcal{L}(\{m_h, s_h, w_{ih}\}_{i,h} | \mathbf{X}) = \sum_t \log \sum_h g_h^t \exp \left[ -\frac{1}{2} \sum_i (r_i^t - y_{ih}^t)^2 \right] \]

\[ y_{ih}^t = w_{ih}^t = \mathbf{v}_{ih} \mathbf{x}_t \]

\[ \Delta \mathbf{v}_{ih} = \eta \sum_t (r_i^t - y_{ih}^t) f_h^t \mathbf{x}_t \]

\[ \Delta m_h = \eta \sum_t (f_h^t - g_h^t) \mathbf{x}_t \]
Competitive MoE: Classification

\[
\mathcal{L}(\{m_h, s_h, w_{ih}\}, i, h | X) = \sum_t \log \sum_h g_h^t \prod_i (y_{ih}^t)^{r_i^t}
\]

\[
= \sum_t \log \sum_h g_h^t \exp \left[ \sum_i r_i^t \log y_{ih}^t \right]
\]

\[
y_{ih}^t = \frac{\exp w_{ih}^t}{\sum_k \exp w_{kh}^t} = \frac{\exp[v_{ih}x^t]}{\sum_k \exp[v_{kh}x^t]}
\]
Hierarchical Mixture of Experts

- Tree of MoE where each MoE is an expert in a higher-level MoE
- **Soft decision tree**: Takes a weighted (gating) average of all leaves (experts), as opposed to using a single path and a single leaf
- Can be trained using EM (Jordan and Jacobs, 1994)