

Lecture 16

Artificial Neural Networks Discussion (3 of 4): Unsupervised Learning and Pattern Recognition

Wednesday, February 23, 2000

William H. Hsu
Department of Computing and Information Sciences, KSU
<http://www.cis.ksu.edu/~bhsu>

Readings:
"The Wake-Sleep Algorithm for Unsupervised Neural Networks", Hinton *et al*
(Reference) Section 6.12, Mitchell
(Reference) Section 3.2.4-3.2.5, Shavlik and Dietterich

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Lecture Outline

- Readings: "The Wake-Sleep Algorithm", Hinton *et al*
- Suggested Reading: 6.12, Mitchell; Rumelhart and Zipser; Kohonen
- This Week's Reviews: Wake-Sleep, Hierarchical Mixtures of Experts
- Unsupervised Learning and Clustering
 - Definitions and framework
 - Constructive induction
 - Feature construction
 - Cluster definition
 - EM, AutoClass, Principal Components Analysis, Self-Organizing Maps
- Expectation-Maximization (EM) Algorithm
 - More on EM and Bayesian Learning
 - EM and unsupervised learning
- Next Lecture: Time Series Learning
 - Intro to time series learning, characterization; stochastic processes
 - Read Chapter 19, Russell and Norvig (neural and Bayesian computation)

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Unsupervised Learning: Objectives

• **Unsupervised Learning**

– Given: data set D

- Vectors of attribute values (x_1, x_2, \dots, x_n)
- No distinction between input attributes and output attributes (class label)

– Return: (synthetic) descriptor y of each x

- Clustering: grouping points (x) into inherent regions of mutual similarity
- Vector quantization: discretizing continuous space with best labels
- Dimensionality reduction: projecting many attributes down to a few
- Feature extraction: constructing (few) new attributes from (many) old ones

Supervised Learning

$x \rightarrow f(x) \rightarrow y$

• **Unsupervised Learning**

– Goal: discover structure in the data

- Organize data into sensible groups (how many here?)
- Criteria: convenient and valid organization of the data
- NB: not necessarily rules for classifying future data points

– Cluster analysis: study of algorithms, methods for discovering this structure

- Representing structure: organizing data into clusters (cluster formation)
- Describing structure: cluster boundaries, centers (cluster segmentation)
- Defining structure: assigning meaningful names to clusters (cluster labeling)

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Clustering

• **A Mode of Unsupervised Learning**


– Given: a collection of data points

– Goal: discover structure in the data

- Organize data into sensible groups (how many here?)
- Criteria: convenient and valid organization of the data
- NB: not necessarily rules for classifying future data points

– Cluster analysis: study of algorithms, methods for discovering this structure

- Representing structure: organizing data into clusters (cluster formation)
- Describing structure: cluster boundaries, centers (cluster segmentation)
- Defining structure: assigning meaningful names to clusters (cluster labeling)



KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Quick Review: Bayesian Learning and EM

- **Problem Definition**
 - Given: data (n -tuples) with missing values, aka partially observable (PO) data
 - Want to fill in ? with expected value
- **Solution Approaches**
 - Expected = distribution over possible values
 - Use "best guess" Bayesian model (e.g., BBN) to estimate distribution
 - Expectation-Maximization (EM) algorithm can be used here
- **Intuitive Idea**
 - Want to find h_{ML} in PO case ($D = \text{unobserved variables} \circ \text{observed variables}$)
 - Estimation step: calculate $E[\text{unobserved variables} | h]$, assuming current h
 - Maximization step: update w_{jk} to maximize $E[\log P(D | h)]$, $D = \text{all variables}$

$$h_{ML} = \arg \max_{h \in H} \frac{\# \text{ data cases with } h, \hat{e}}{\# \text{ data cases with } \hat{e}} = \arg \max_{h \in H} \frac{\sum_{\substack{N: h, \hat{e} \in \\ \sum_j I_{\hat{e} = \hat{e}}(\hat{X}_j)}} \binom{N}{h, \hat{e}}}{\sum_j I_{\hat{e} = \hat{e}}(\hat{X}_j)}$$

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

EM for Unsupervised Learning


- **Unsupervised Learning Problem**
 - Objective: estimate a probability distribution with unobserved variables
 - Use EM to estimate mixture policy (more on this later; see 6.12, Mitchell)
- **Pattern Recognition Examples**
 - Human-computer intelligent interaction (HCII)
 - Detecting facial features in emotion recognition
 - Gesture recognition in virtual environments
 - Computational medicine [Frey, 1998]
 - Determining morphology (shapes) of bacteria, viruses in microscopy
 - Identifying cell structures (e.g., nucleus) and shapes in microscopy
 - Other image processing
 - Many other examples (audio, speech, signal processing; motor control; etc.)
- **Inference Examples**
 - Plan recognition: mapping from (observed) actions to agent's (hidden) plans
 - Hidden changes in context: e.g., aviation; computer security; MUDs

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Unsupervised Learning: Competitive Learning for Feature Discovery


- Intuitive Idea: Competitive Mechanisms for Unsupervised Learning**
 - Global organization from local, competitive weight update
 - Basic principle expressed by Von der Malsburg
 - Guiding examples from (neuro)biology: lateral inhibition
 - Previous work: Hebb, 1949; Rosenblatt, 1959; Von der Malsburg, 1973; Fukushima, 1975; Grossberg, 1976; Kohonen, 1982
- A Procedural Framework for Unsupervised Connectionist Learning**
 - Start with identical ("neural") processing units, with random initial parameters
 - Set limit on "activation strength" of each unit
 - Allow units to compete for right to respond to a set of inputs
- Feature Discovery**
 - Identifying (or *constructing*) new features relevant to supervised learning
 - Examples: finding distinguishable letter characteristics in handwritten character recognition (HCR), optical character recognition (OCR)
 - Competitive learning: transform X into X' ; train units in X' closest to x



CIS 830: Advanced Topics in Artificial Intelligence Kansas State University
Department of Computing and Information Sciences

Unsupervised Learning: Kohonen's Self-Organizing Map (SOM) [1]

- Another Clustering Algorithm**
 - aka Self-Organizing Feature Map (SOFM)
 - Given: vectors of attribute values (x_1, x_2, \dots, x_n)
 - Returns: vectors of attribute values $(x'_1, x'_2, \dots, x'_k)$
 - Typically, $n \gg k$ (n is high, $k = 1, 2, \text{ or } 3$; hence "dimensionality reducing")
 - Output: vectors x' , the projections of input points x ; also get $P(x'_j | x)$
 - Mapping from x to x' is topology preserving
- Topology Preserving Networks**
 - Intuitive idea: similar input vectors will map to similar clusters
 - Recall: informal definition of cluster (isolated set of mutually similar entities)
 - Restatement: "clusters of X (high-D) will still be clusters of X' (low-D)"
- Representation of Node Clusters**
 - Group of neighboring artificial neural network units (neighborhood of nodes)
 - SOMs: combine ideas of topology-preserving networks, unsupervised learning
- Implementation**: <http://www.cis.hut.fi/nrc/> and MATLAB NN Toolkit




CIS 830: Advanced Topics in Artificial Intelligence Kansas State University
Department of Computing and Information Sciences

Unsupervised Learning: Kohonen's Self-Organizing Map (SOM) [2]

- Kohonen Network (SOM) for Clustering**
 - Training algorithm: unnormalized competitive learning
 - Map is organized as a grid (shown here in 2D)
 - Each node (grid element) has a weight vector w_j
 - Dimension of w_j is n (same as input vector)
 - Number of trainable parameters (weights): $m \cdot m \cdot n$ for an m -by- m SOM
 - 1999 state-of-the-art: typical small SOMs 5-20, "industrial strength" > 20
 - Output found by selecting j^* whose w_j has minimum Euclidean distance from x
 - Only one active node, aka Winner-Take-All (WTA): winning node j^*
 - i.e., $j^* = \arg \min_j \|w_j - x\|_2$
 - Update Rule**
 - Same as competitive learning algorithm, with one modification
 - Neighborhood function associated with j^* spreads the w_j around


$$\dot{w}_j(t+1) = \begin{cases} \dot{w}_j(t) + r(t)h_{j,j^*}(x - \dot{w}_j(t)) & \text{if } j \in \text{Neighborhood}(j^*) \\ \dot{w}_j(t) & \text{otherwise} \end{cases}$$



CIS 830: Advanced Topics in Artificial Intelligence Kansas State University
Department of Computing and Information Sciences

Unsupervised Learning: Kohonen's Self-Organizing Map (SOM) [3]

- Traditional Competitive Learning**
 - Only train j^*
 - Corresponds to neighborhood of 0
- Neighborhood Function h_{j,j^*}**
 - For 2D Kohonen SOMs, h is typically a square or hexagonal region
 - j^* , the winner, is at the center of Neighborhood (j^*)
 - $h_{j,j^*} = 1$
 - Nodes in Neighborhood (j) updated whenever j wins, i.e., $j^* = j$
 - Strength of information fed back to w_j is inversely proportional to its distance from the j^* for each x
 - Often use exponential or Gaussian (normal) distribution on neighborhood to decay weight delta as distance from j^* increases
- Annealing of Training Parameters**
 - Neighborhood must shrink to 0 to achieve convergence
 - r (learning rate) must also decrease monotonically



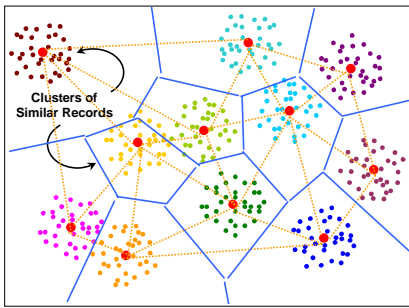
CIS 830: Advanced Topics in Artificial Intelligence Kansas State University
Department of Computing and Information Sciences

Unsupervised Learning: SOM and Other Projections for Clustering


Dimensionality-Reducing Projection (x')

Delaunay Triangulation

Voronoi (Nearest Neighbor) Diagram (j)




Cluster Formation and Segmentation Algorithm (Sketch)



CIS 830: Advanced Topics in Artificial Intelligence Kansas State University
Department of Computing and Information Sciences

Unsupervised Learning: Other Algorithms (PCA, Factor Analysis)

- Intuitive Idea**
 - Q: *Why are dimensionality-reducing transforms good for supervised learning?*
 - A: There may be many attributes with undesirable properties, e.g.,
 - Irrelevance: x_i has little discriminatory power over $c(x) = y_i$
 - Sparseness of information: "feature of interest" spread out over many x_i 's (e.g., text document categorization, where x_i is a word position)
 - We want to increase the "information density" by "squeezing X down"
- Principal Components Analysis (PCA)**
 - Combining redundant variables into a single variable (aka component, or factor)
 - Example: ratings (e.g., Nielsen) and polls (e.g., Gallup); responses to certain questions may be correlated (e.g., "like fishing?" "time spent boating")
- Factor Analysis (FA)**
 - General term for a class of algorithms that includes PCA
 - Tutorial: <http://www.statsoft.com/textbook/stfacan.html>



CIS 830: Advanced Topics in Artificial Intelligence Kansas State University
Department of Computing and Information Sciences


Clustering Methods: Design Choices

- Intuition**
 - Functional (declarative) definition: easy ("We recognize a cluster when we see it")
 - Operational (procedural, constructive) definition: much harder to give
 - Possible reason: clustering of objects into groups has taxonomic semantics (e.g., shape, size, time, resolution, etc.)
- Possible Assumptions**
 - Data generated by a particular probabilistic model
 - No statistical assumptions
- Design Choices**
 - Distance (similarity) measure: standard metrics, transformation-invariant metrics
 - L_1 (Manhattan): $\sum |x_i - y_i|$, L_2 (Euclidean): $\sqrt{\sum (x_i - y_i)^2}$, L_∞ (Sup): $\max |x_i - y_i|$
 - Symmetry: Mahalanobis distance
 - Shift, scale invariance: covariance matrix
 - Transformations (e.g., covariance diagonalization): rotate axes to get rotational invariance, cf. PCA, FA)

KSU
Kansas State University
Department of Computing and Information Sciences


CIS 830: Advanced Topics in Artificial Intelligence

Clustering: Applications



NCSA D2K 1.0 - <http://www.ncsa.uiuc.edu/STI/ALG/>


Transactional Database Mining



Data from T. Mitchell's web site:
<http://www.cs.cmu.edu/~tom/faces.html>

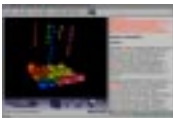
<http://www.cnl.salk.edu/~wisotto/Biographies/FaceFeatureFinding.html>

Facial Feature Extraction



5500 news stories from the WWW in 1997

ThemeScapes - <http://www.cartia.com>



NCSA D2K 2.0 - <http://www.ncsa.uiuc.edu/STI/ALG/>

Confidential and proprietary to Caterpillar; may only be used with prior written consent from Caterpillar.

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Unsupervised Learning and Constructive Induction

- Unsupervised Learning in Support of Supervised Learning**
 - Given: $D =$ labeled vectors (x, y)
 - Return: $D' =$ transformed training examples (x', y')
 - Solution approach: constructive induction
 - Feature "construction": generic term
 - Cluster definition
- Feature Construction: Front End**
 - Synthesizing new attributes
 - Logical: $x_1 \vee \dots \vee x_2$, arithmetic: $x_1 + x_2 / x_2$
 - Other synthetic attributes: $f(x_1, x_2, \dots, x_n)$, etc.
 - Dimensionality-reducing projection, feature extraction
 - Subset selection: finding relevant attributes for a given target y
 - Partitioning: finding relevant attributes for given targets y_1, y_2, \dots, y_p
- Cluster Definition: Back End**
 - Form, segment, and label clusters to get intermediate targets y'
 - Change of representation: find an (x', y') that is good for learning target y

Constructive Induction
(x, y)

↓

Feature (Attribute)
Construction and
Partitioning

↓

$x' = f(x_1, \dots, x_n)$

↓

Cluster
Definition

↓

(x', y') or $((x'_1, y'_1), \dots, (x'_p, y'_p))$

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Clustering: Relation to Constructive Induction

- Clustering versus Cluster Definition**
 - Clustering: 3-step process
 - Cluster definition: "back end" for feature construction
- Clustering: 3-Step Process**
 - Form
 - (x'_1, \dots, x'_k) in terms of (x_1, \dots, x_n)
 - NB:** typically part of construction step, sometimes integrates both
 - Segment
 - (y'_1, \dots, y'_j) in terms of (x'_1, \dots, x'_k)
 - NB:** number of clusters J not necessarily same as number of dimensions k
 - Label
 - Assign names (discrete/symbolic labels (v'_1, \dots, v'_j)) to (y'_1, \dots, y'_j)
 - Important in document categorization (e.g., clustering text for info retrieval)
- Hierarchical Clustering: Applying Clustering Recursively**

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Terminology

- Expectation-Maximization (EM) Algorithm**
 - Iterative refinement: repeat until convergence to a locally optimal label
 - Expectation step: estimate parameters with which to simulate data
 - Maximization step: use simulated ("fictitious") data to update parameters
- Unsupervised Learning and Clustering**
 - Constructive induction: using unsupervised learning for supervised learning
 - Feature construction: "front end" - construct new x values
 - Cluster definition: "back end" - use these to reformulate y
 - Clustering problems: formation, segmentation, labeling
 - Key criterion: distance metric (points closer intra-cluster than inter-cluster)
 - Algorithms
 - AutoClass: Bayesian clustering
 - Principal Components Analysis (PCA), factor analysis (FA)
 - Self-Organizing Maps (SOM): topology preserving transform (dimensionality reduction) for competitive unsupervised learning

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence

Summary Points

- Expectation-Maximization (EM) Algorithm**
- Unsupervised Learning and Clustering**
 - Types of unsupervised learning
 - Clustering, vector quantization
 - Feature extraction (typically, dimensionality reduction)
 - Constructive induction: unsupervised learning in support of supervised learning
 - Feature construction (aka feature extraction)
 - Cluster definition
 - Algorithms
 - EM: mixture parameter estimation (e.g., for AutoClass)
 - AutoClass: Bayesian clustering
 - Principal Components Analysis (PCA), factor analysis (FA)
 - Self-Organizing Maps (SOM): projection of data; competitive algorithm
 - Clustering problems: formation, segmentation, labeling
- Next Class: Presentation on Modular and Hierarchical ANNs**

KSU
Kansas State University
Department of Computing and Information Sciences

CIS 830: Advanced Topics in Artificial Intelligence