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

Readings:
"The Wake-Sleep Algorithm for Unsupervised Neural Networks", Hinton et al (Reference) Section 6.12, Mitchell
"A Mode of Unsupervised Learning"

Unsupervised Learning: Objectives

- Unsupervised Learning
  - Given: data set D
  - Objectives
    - 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
- Intuitive Idea
  - Want to map independent variables (x) to dependent variables (y = f(x))
  - Don’t always know what “dependent variables” (y) are
  - Need to discover y based on numerical criterion (e.g., distance metric)

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)
- Cluster: Informal and Formal Definitions
  - Set whose entities are alike and are different from entities in other clusters
  - Aggregation of points in the instance space such that distance between any two points in the cluster is less than the distance between any point in the cluster and any point not in it

Quick Review: Bayesian Learning and EM

- Problem Definition
  - Given: data (n-tuples) with missing values, aka partially observable (PO) data
  - Want to fill in a with expected values
- 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 $h_\theta$ to maximize $\log p(D | h)$, assuming current $h$
  - Maximization step: update $w_k$ to maximize $\log p(D | h, D \sim \theta)$
- Inference Examples
  - Plan recognition: mapping from (observed) actions to agent’s (hidden) plans
  - Hidden changes in context: e.g., aviation; computer security; MIDs

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 (HCI)
  - Detecting facial features in emotion recognition
  - Gesture recognition in virtual environments
  - Computational medicine [Frey, 1998]
  - Determining morphology (shapes) of bacteria, viruses in microscopy
- Organize data into sensible groups (how many here?)

Lecture Outline

- Suggested Readings: 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)
Unsupervised Learning: Competitive Mechanisms for Unsupervised Learning

- Intuitive Idea: Competitive Mechanisms for Unsupervised Learning
  - Global organization from local, competitive weight update
  - 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 (OCR), optical character recognition (OCR)
  - Competitive learning: transforms X into X'; train units in X closest to x

Number of trainable parameters (weights): m

- Dimension of w j is n  (same as input vector)
- Each node (grid element) has a weight vector w j

Guiding examples from (neuro)biology: lateral inhibition

Basic principle expressed by Von der Malsburg

- Neighborhood function h j, j*

Annealing of Training Parameters

- Neighborhood must shrink to 0 to achieve convergence
- Often use exponential or Gaussian (normal) distribution on neighborhood to decay weight delta as distance from j* increases
- Annaling of Training Parameters
  - Neighborhood must shrink to 0 to achieve convergence
  - r (learning rate) must also decrease monotonically

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

- 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

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 · n for an m-by-n 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||

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

- Another Clustering Algorithm
  - aka Self-Organizing Feature Map (SOFM)
  - Given: vectors of attribute values (x 1 , x 2 , …, x k )
  - Returns: vectors of attribute values (x 1 ', x 2 ', …, x k ')
  - Typically, n >> k (n is high, k = 1, 2, or 3; hence “dimensionality reducing”)
  - Output: vectors x'; the projections of input points x; also get P(x); (x)
  - Mapping from x to x’ in topology preserving

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 has little discriminatory power over c(x) = y;
      - Sparseness of information: “feature of interest” spread out over many x’s (e.g., text document categorization, where x 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)
  - Examples: 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
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
    - Clustering definition
  - Feature Construction: Front End
    - Synthesizing new attributes
      - Logical: \( x_1 \land x_2 \), arithmetic: \( x_1 \times x_2 \)
      - Other synthetic attributes: \( f(x_1, x_2, ..., x_n) \), etc.
    - Dimensionality-reducing projection, feature extraction
    - Sub-set selection: finding relevant attributes for a given target y
    - Partitioning: finding relevant attributes for given targets \( y_1, y_2, ..., y_N \)
  - 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**
  - Clustering versus Cluster Definition
    - Clustering: 3-step process
      - Cluster definition: “back end” for feature construction
    - Clustering: 3-Step Process
      - Form
        - \((x', y')\) in terms of \((x_1, y_1)\)
        - NB: typically part of construction step, sometimes integrates both
      - Segment
        - \((y_1', ..., y_J')\) in terms of \((x_1, y_1)\)
        - NB: number of clusters \( J \) not necessarily same as number of dimensions \( k \)
      - Label
        - Assign names (discrete/symbolic labels \((y_1', ..., y_J')\)) to \((y_1', ..., y_J')\)
        - Important in document categorization (e.g., clustering text for info retrieval)
  - Hierarchical Clustering: Applying Clustering Recursively

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 into 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 unsupervised learning

Summary Points

- **Expectation-Maximization (EM) Algorithm**
  - Iterative refinement: repeat until convergence to a locally optimal label
- **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
  - Next Class: Presentation on Modular and Hierarchical ANNs