

Instance-Based Learning (IBL): *k*-Nearest Neighbor and Radial Basis Functions

Tuesday, November 20, 2001

William H. Hsu Department of Computing and Information Sciences, KSU

http://www.cis.ksu.edu/~bhsu

Readings: Chapter 8, Mitchell



CIS 732: Machine Learning and Pattern Recognition

Lecture Outline

- Readings: Chapter 8, Mitchell
- Suggested Exercises: 8.3, Mitchell
- Next Week's Paper Review (Last One!)
 - "An Approach to Combining Explanation-Based and Neural Network Algorithms", Shavlik and Towell
 - Due Tuesday, 11/30/1999
- k-Nearest Neighbor (k-NN)
 - IBL framework
 - IBL and case-based reasoning
 - Prototypes
 - Distance-weighted k-NN
- Locally-Weighted Regression
- Radial-Basis Functions
- Lazy and Eager Learning
- Next Lecture (Tuesday, 11/30/1999): Rule Learning and Extraction



Instance-Based Learning (IBL)

- Intuitive Idea
 - Store all instances <x, c(x)</p>
 - Given: <u>query instance</u> x_q
 - Return: function (e.g., label) of closest instance in database of *prototypes*
 - Rationale
 - Instance closest to x_q tends to have target function close to $f(x_q)$
 - Assumption can fail for deceptive hypothesis space or with too little data!
- Nearest Neighbor
 - First locate nearest training example x_n to query x_q
 - Then estimate $\hat{f} x_q ? f x_n$?
- k-Nearest Neighbor
 - Discrete-valued f: take vote among k nearest neighbors of x_q
 - Continuous-valued f:

$$\hat{f}_{\mathbf{x}_{q}}^{?}$$
? $\frac{?_{i_{2}}^{k}f_{\mathbf{x}_{i}}^{?}}{k}$



When to Consider Nearest Neighbor

- Ideal Properties
 - Instances map to points in Rⁿ
 - Fewer than 20 attributes per instance
 - Lots of training data
- Advantages
 - Training is very fast
 - Learn complex target functions
 - Don't lose information
- Disadvantages
 - Slow at query time
 - Easily fooled by irrelevant attributes



Voronoi Diagram





CIS 732: Machine Learning and Pattern Recognition

k-NN and Bayesian Learning: Behavior in the Limit

- Consider: Probability Distribution over Labels
 - Let *p* denote learning agent's *belief* in the distribution of labels
 - p(x)? probability that instance x will be labeled 1 (positive) versus 0 (negative)
 - <u>Objectivist view</u>: as more evidence is collected, approaches "true probability"
- Nearest Neighbor
 - As number of training examples? ? , approaches behavior of Gibbs algorithm
 - Gibbs: with probability p(x) predict 1, else 0
- k-Nearest Neighbor
 - As number of training examples ? ? and *k* gets large, approaches Bayes optimal
 - Bayes optimal: if p(x) > 0.5 then predict 1, else 0
- Recall: Property of Gibbs Algorithm
 - Elerror $h_{Gibbs} ?? 2E'_{error} h_{BayesOptim al} ??$
 - Expected error of Gibbs no worse than twice that of Bayes optimal



Distance-Weighted *k*-NN

- Intuitive Idea
 - Might want to weight *nearer* neighbors more heavily
 - Rationale
 - Instances closer to x_q tend to have target functions closer to $f(x_q)$
 - Want benefit of BOC over Gibbs (k-NN for large k over 1-NN)
- Distance-Weighted Function

$$\hat{f}_{i}^{2} \mathbf{x}_{q}^{2}? \quad \frac{? \sum_{i \geq 1}^{k} \mathbf{w}_{i} ? f ? \mathbf{x}_{i} ?}{? \sum_{i \geq 1}^{k} \mathbf{w}_{i}}$$

o distance:
$$\mathbf{w}_{i} ? \frac{1}{d ? \mathbf{x}_{q}, \mathbf{x}_{i} ?}$$

- Weights are proportional to distance:
- $d(x_q, x_i)$ is Euclidean distance
- NB: now it makes sense to use all $\langle x, f(x) \rangle$ instead of just k? Shepard's method
- Jargon from Statistical Pattern Recognition
 - <u>Regression</u>: approximating a real-valued target function
 - <u>Residual</u>: error f?x?? f?x?
 - <u>Kernel function</u>: function K such that $w_i ? K d x_q, x_i$



CIS 732: Machine Learning and Pattern Recognition

Curse of Dimensionality

- A Machine Learning Horror Story
 - Suppose
 - Instances described by *n* attributes $(x_1, x_2, ..., x_n)$, e.g., n = 20
 - Only *n'* << *n* are relevant, e.g., *n*' = 2
 - Horrors! Real KDD problems usually *are* this bad or *worse*... (correlated, etc.)
 - <u>Curse of dimensionality</u>: nearest neighbor learning algorithm is easily mislead when *n* large (i.e., <u>high-dimension</u> *X*)
- Solution Approaches
 - *Dimensionality reducing transformations* (e.g., SOM, PCA; see Lecture 15)
 - Attribute weighting and attribute subset selection
 - Stretch *j*th axis by weight z_j : $(z_1, z_2, ..., z_n)$ chosen to minimize prediction error
 - Use cross-validation to automatically choose weights $(z_1, z_2, ..., z_n)$
 - *NB*: setting *z_i* to 0 eliminates this dimension altogether
 - See [Moore and Lee, 1994; Kohavi and John, 1997]



Locally Weighted Regression

- Global versus Local Methods
 - <u>Global</u>: consider all training examples $\langle x, f(x) \rangle$ when estimating $f(x_q)$
 - Local: consider only examples within local neighborhood (e.g., *k* nearest)
- Locally Weighted Regression
 - Local method
 - Weighted: contribution of each training example is weighted by distance from x_q
 - Regression: approximating a real-valued target function
- Intuitive Idea
 - k-NN forms local approximation to f(x) for each x_q
 - Explicit approximation to f(x) for region surrounding x_q
 - Fit parametric function \hat{f} ??: e.g., linear, quadratic (piecewise approximation)
- Choices of Error to Minimize
 - <u>Sum squared error (SSE) over k-NN</u>

 $E_1?x_q??\frac{1}{2}?.\frac{1}{2}?!f?x??\hat{f}?x??\hat{f}?x?!$

- Distance-weighted SSE over *all* neighbors

$$E_{2}?x_{q}??\frac{1}{2}?_{x^{2}D}f?x??\hat{f}?x??\hat{f}?x??$$



Kansas State University Department of Computing and Information Sciences

<u>Radial Basis Function (RBF) Networks</u>

- What Are RBF Networks?
 - Global approximation to target function f, in terms of linear combination of local _ approximations
 - Typical uses: image, signal classification _
 - Different kind of artificial neural network (ANN)
 - Closely related to distance-weighted regression, but "eager" instead of "lazy"
- **Activation Function**



CIS 732: Machine Learning and Pattern Recognition

RBF Networks: Training

- Issue 1: Selecting Prototypes
 - What x_u should be used for each kernel function $K_u(d(x_u, x))$
 - Possible prototype distributions
 - Scatter uniformly throughout instance space
 - Use training instances (reflects instance distribution)
- Issue 2: Training Weights
 - Here, assume Gaussian K_u
 - First, choose hyperparameters
 - Guess variance, and perhaps mean, for each K_u
 - e.g., use EM
 - Then, hold K_u fixed and train parameters
 - Train weights in linear output layer
 - Efficient methods to fit linear function



Case-Based Reasoning (CBR)

- Symbolic Analogue of Instance-Based Learning (IBL)
 - Can apply IBL even when X? Rⁿ
 - Need different "distance" metric
 - Intuitive idea: use symbolic (e.g., syntactic) measures of similarity
- Example
 - Declarative knowledge base
 - Representation: symbolic, logical descriptions
 - ((user-complaint rundll-error-on-shutdown) (system-model thinkpad-600-E) (cpu-model mobile-pentium-2) (clock-speed 366) (network-connection PC-MCIA-100-base-T) (memory 128-meg) (operating-system windows-98) (installed-applications office-97 MSIE-5) (disk-capacity 6-gigabytes))
 - (likely-cause ?)



Case-Based Reasoning in CADET

- CADET: CBR System for Functional Decision Support [Sycara et al, 1992]
 - 75 stored examples of mechanical devices
 - Each training example: <qualitative function, mechanical structure>
 - <u>New query</u>: desired function
 - Target value: mechanical structure for this function
- Distance Metric
 - Match qualitative functional descriptions
 - X? Rⁿ, so "distance" is not Euclidean even if it is quantitative



CADET: Example

- **Stored Case: T-Junction Pipe** ۲
 - **Diagrammatic** knowledge _
 - Structure, function _





Function





- **Problem Specification: Water Faucet** ullet
 - **Desired function:**



Structure: ?



CADET: Properties

- Representation
 - Instances represented by rich structural descriptions
 - Multiple instances retreived (and combined) to form solution to new problem
 - Tight coupling between case retrieval and new problem
- Bottom Line
 - Simple matching of cases useful for tasks such as answering help-desk queries
 - Compare: technical support knowledge bases
 - Retrieval issues for natural language queries: not so simple...
 - User modeling in web IR, interactive help)
 - Area of continuing research



Lazy and Eager Learning

- Lazy Learning
 - Wait for query before generalizing
 - Examples of lazy learning algorithms
 - k-nearest neighbor (k-NN)
 - <u>Case-based reasoning (CBR)</u>
- Eager Learning
 - Generalize before seeing query
 - Examples of eager learning algorithms
 - <u>Radial basis function (RBF) network training</u>
 - ID3, backpropagation, simple (Naïve) Bayes, etc.
- Does It Matter?
 - Eager learner must create global approximation
 - Lazy learner can create many local approximations
 - If they use same H, lazy learner can represent more complex functions
 - e.g., consider H? linear functions





Terminology

- <u>Instance Based Learning (IBL)</u>: Classification Based On Distance Measure
 - k-Nearest Neighbor (k-NN)
 - Voronoi diagram of order k: data structure that answers k-NN queries x_q
 - Distance-weighted k-NN: weight contribution of k neighbors by distance to x_{a}
 - Locally-weighted regression
 - Function approximation method, generalizes *k*-NN
 - Construct explicit approximation to target function f(?) in neighborhood of x_q
 - <u>Radial-Basis Function (RBF) networks</u>
 - Global approximation algorithm
 - Estimates linear combination of local kernel functions
- <u>Case-Based Reasoning (CBR)</u>
 - Like IBL: lazy, classification based on similarity to prototypes
 - Unlike IBL: similarity measure not necessarily distance metric
- Lazy and Eager Learning
 - <u>Lazy</u> methods: may consider query instance x_q when generalizing over D
 - <u>Eager</u> methods: choose global approximation *h* before x_q observed



Summary Points

- <u>Instance Based Learning (IBL)</u>
 - k-Nearest Neighbor (k-NN) algorithms
 - When to consider: few continuous valued attributes (low dimensionality)
 - Variants: distance-weighted k-NN; k-NN with attribute subset selection
 - Locally-weighted regression: function approximation method, generalizes *k*-NN
 - <u>Radial-Basis Function (RBF) networks</u>
 - Different kind of artificial neural network (ANN)
 - Linear combination of local approximation ? global approximation to f(?)
- <u>Case-Based Reasoning (CBR) Case Study: CADET</u>
 - Relation to IBL
 - CBR online resource page: <u>http://www.ai-cbr.org</u>
- Lazy and Eager Learning
- Next Week
 - Rule learning and extraction
 - <u>Inductive logic programming (ILP)</u>



