Dr. George Karraz, Ph. D.

Machine Learning
k-nearest neighbor

Dr. George Karraz, Ph. D.

A classification of learning algorithms

- Eager learning algorithms
  - Neural networks
  - Decision trees
  - Bayesian classifiers
- Lazy learning algorithms
  - K-nearest neighbor
  - Case based reasoning

General Idea of Instance-based Learning

- Learning: store all the data instances
- Performance:
  - when a new query instance is encountered
    - retrieve a similar set of related instances from memory
    - use to classify the new query

Pros and Cons of Instance Based Learning

- Pros
  - Can construct a different approximation to the target function for each distinct query instance to be classified
  - Can use more complex, symbolic representations
- Cons
  - Cost of classification can be high
  - Uses all attributes (do not learn which are most important)

k-nearest neighbor (knn) learning

- Most basic type of instance learning
- Assumes all instances are points in n-dimensional space
- A distance measure is needed to determine the “closeness” of instances
- Classify an instance by finding its nearest neighbors and picking the most popular class among the neighbors

Important Decisions

- Distance measure
- Value of k (usually odd)
- Voting mechanism
- Memory indexing

Euclidean Distance

- Typically used for real valued attributes
- Instance x (often called a feature vector)

$$< a_1(x), a_2(x), \ldots a_n(x) >$$

- Distance between two instances x_i and x_j

$$d(x_i, x_j) \equiv \sqrt{\sum_{r=1}^{n} (a_r(x_i) - a_r(x_j))^2}$$

Discrete Valued Target Function

Training algorithm:

For each training example <x, f(x), add the example to the list *training examples*

Classification algorithm:

Given a query instance x_q to be classified

Let $x_1 \ldots x_k$ be the k training examples nearest to x_q

Return

$$\hat{f}(x_q) \leftarrow \arg\max_{v \in V} \sum_{i=1}^{k} \delta(v, f(x_i))$$

where $\delta(a,b) = 1$ if $a = b$

$\delta(a,b) = 0$ otherwise

Continuous valued target function

- Algorithm computes the mean value of the k nearest training examples rather than the most common value
- Replace fine line in previous algorithm with

$$\hat{f}(x_q) \leftarrow \frac{\sum_{i=1}^{k} f(x_i)}{k}$$

Training dataset

Customer ID	Debt	Income	Marital Status	Risk
Abel	High	High	Married	Good
Ben	Low	High	Married	Doubtful
Candy	Medium	Very low	Unmarried	Poor
Dale	Very high	Low	Married	Poor
Ellen	High	Low	Married	Poor
Fred	High	Very low	Married	Poor
George	Low	High	Unmarried	Doubtful
Harry	Low	Medium	Married	Doubtful
Igor	Very Low	Very High	Married	Good
Jack	Very High	Medium	Married	Poor

k-nn

- K = 3
- Distance
  - Score for an attribute is 1 for a match and 0 otherwise
  - Distance is sum of scores for each attribute
- Voting scheme: proportionate voting in case of ties

Test Set

Customer ID	Debt	Income	Marital Status	Risk
Zeb	High	Medium	Married	?
Yong	Low	High	Married	?
Xu	Very low	Very low	Unmarried	?
Vasco	High	Low	Married	?
Unace	High	Low	Divorced	?
Trey	Very low	Very low	Married	?
Steve	Low	High	Unmarried	?

Algorithm questions

- What is the space complexity for the model?
- What is the time complexity for learning the model?
- What is the time complexity for classification of an instance?