CSE4334/5334 Data Mining
K-NN Classification
(Put together from many sources)

Sharma Chakravarthy

Department of Computer Science and Engineering
University of Texas at Arlington
Fall 2019 (acknowledgement to Pang-Ning Tan, Michael Steinbach and Vipin Kumar, and Jiawei Han, Micheline Kamber and Jian Pei)

K-NN Introduction

- K-nearest neighbors (KNN) algorithm is a type of supervised ML algorithm which can be used for both classification as well as regression predictive problems. However, it is mainly used for classification problems in industry. The following two properties define KNN
- **Lazy learning algorithm** − KNN is a lazy learning algorithm because it does not have a specialized training phase and uses all training data while classification (no model is generated!)
- **Non-parametric learning algorithm** − KNN is also a non-parametric learning algorithm because it doesn't assume anything about the underlying data.

2

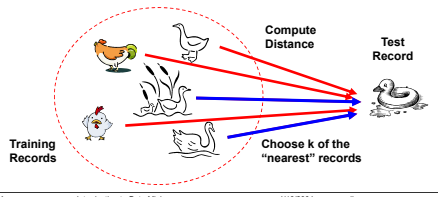
Working of K-NN algorithm

- K-nearest neighbors (KNN) algorithm uses 'feature similarity' to predict the class of new data points
- This means that the new data point will be assigned a value/class based on how closely it matches the points in the training set.
- We can understand its working with the help of the following

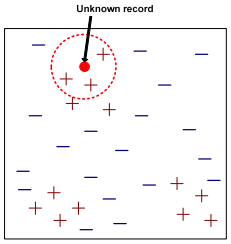
3

Nearest Neighbor Classifiers

- Basic idea:
  - If it walks like a duck, quacks like a duck, then it's probably a duck

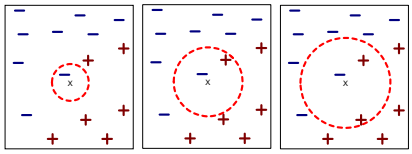


Nearest-Neighbor Classifiers



- Require three things
  - The set of known records
  - Distance Metric to compute distance between records
  - The value of *k*, the number of nearest neighbors to retrieve
- To classify an unknown record:
  - Compute distance to other training records
  - Identify *k* nearest neighbors
  - Use class labels of nearest neighbors to determine the class label of unknown record (e.g., by taking majority vote)

Definition of Nearest Neighbor



(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

K-nearest neighbors of a record x are data points that have the k smallest distance to x

Working of K-NN algorithm

- **Step 1** − For K-NN algorithm, we need labeled data set. So during the first step of KNN, we must load the training as well as test data.
- **Step 2** − Next, we need to choose the value of K, i.e., the nearest data points. K can be any integer.
- **Step 3** − For each point in the test data do the following −
  - **3.1** − Calculate the distance between test data and each training data with the help of any of the method namely: Euclidean, Manhattan or Hamming distance. The most commonly used method to calculate distance is Euclidean.
  - **3.2** − Now, based on the distance value, sort them in ascending order.
  - **3.3** − Next, it will choose the top K closest from the sorted array.
  - **3.4** − Now, it will assign a class to the test point based on most frequent class of these rows.
- **Step 4** − End

Pros and cons of K-NN

- Pros
  - It is a very simple algorithm to understand and interpret.
  - It is very useful for nonlinear data because there is no assumption about data in this algorithm.
  - It is a versatile algorithm as we can use it for classification as well as regression.
  - It has relatively high accuracy but there are much better supervised learning models than KNN.

Pros and cons of K-NN

- Cons
  - It is computationally a bit expensive because it computes similarity with all the training data.
  - High memory storage required as compared to other supervised learning algorithms.
  - Prediction is slow in case of big N.
  - It is very sensitive to the scale of data as well as irrelevant features.

Applications of K-NN

- Banking System
  - KNN can be used in banking system to predict weather an individual is fit for loan approval? Does that individual have the characteristics similar to the defaulters?
- Calculating Credit Ratings
  - KNN algorithms can be used to find an individual's credit rating by comparing with the persons having similar traits.

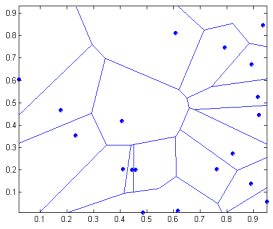
Applications of K-NN

- Politics
  - With the help of KNN algorithms, we can classify a potential voter into various classes like "Will Vote", "Will not Vote", "Will Vote to Party 'democrat', "Will Vote to Party 'republican'.
- Other areas in which KNN algorithm can be used are Speech Recognition, Handwriting Detection, Image Recognition and Video Recognition.

1 nearest-neighbor

Voronoi Diagram

In mathematics, a Voronoi diagram is a partitioning of a plane into regions based on distance to points in a specific subset of the plane. That set of points is specified beforehand, and for each seed there is a corresponding region consisting of all points closer to that seed than to any other.



Nearest Neighbor Classification

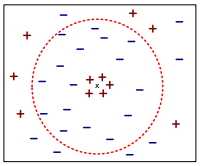
- Compute distance between two points:
  - Euclidean distance

$$d(p,q) = \sqrt{\sum_i (p_i - q_i)^2}$$

- Determine the class from nearest neighbor list
  - take the majority vote of class labels among the k-nearest neighbors
  - Weigh the vote according to distance
    - weight factor, $w = 1/d^2$

Nearest Neighbor Classification…

- Choosing the value of k:
  - If k is too small, sensitive to noise points (may over fit)
  - If k is too large, neighborhood may include points from other classes (misclassification)



Nearest Neighbor Classification…

- Scaling issues
  - Attributes may have to be scaled to prevent distance measures from being dominated by one of the attributes
  - Example:
    - height of a person may vary from 1.5m to 1.8m
    - weight of a person may vary from 90lb to 300lb
    - income of a person may vary from $10K to $1M

Nearest Neighbor Classification…

- Problem with Euclidean measure:
  - High dimensional data
    - curse of dimensionality
  - Can produce counter-intuitive results

1 1 1 1 1 1 1 1 1 1 1 0		1 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 1 1 1	vs	0 0 0 0 0 0 0 0 0 0 0 1
d = 1.4142		d = 1.4142

- Solution: Normalize the vectors to unit length

Nearest neighbor Classification…

- k-NN classifiers are lazy learners
  - It does not build models explicitly
  - Unlike eager learners such as decision tree induction and rule-based systems
  - Classifying unknown records are relatively expensive (why?)
    - The amount of work at test time is little for decision trees as the model has been generated.
    - Here for each test data, you have to recompute distances to all points.

Nearest neighbor Classification…

- k-NN classification is part of a more general technique known as instance-based learning
- No need for model building; hence, expensive as well
- K-NN classifiers make prediction based on local information in contrast to DT use a global model
- K-NN classifiers can produce decision boundaries of arbitrary shape. Provides more flexibility (DTs are constrained to rectilinear decision boundaries)
- Missing values may be a problem as distance requires all attribute values
- Presence of irrelevant attributes can affect K-NN classifiers
- Choice of proximity measure is very important! Otherwise, may produce wrong predictions

Ensemble classifiers

- Also known as classifier combination methods
  - Constructs a set of base classifiers from training data and performs classification by taking a vote on the prediction made by each classifier
  - They tend to perform better than any single classifier
- Example
  - Consider 25 binary classifiers, each with an error rate of 0.35
  - If all give the same result for a test case, the error is still 0.35 even with majority voting
  - If the classifiers are *independent (i.e., their error rates are uncorrelated), the error reduces to 0.06! (see the formula in the text book)*
- Ensemble classifiers are constructed in several ways

Thank You !!!

For more information visit:
http://itlab.uta.edu