CS 4602

Introduction to Machine Learning

K-Nearest Neighbor

Instructor: Po-Chih Kuo

Roadmap

- Introduction and Basic Concepts
- Regression
- Bayesian Classifiers
- Decision Trees
- Linear Classifier
- Neural Networks
- Deep learning
- Convolutional Neural Networks
- The others
- KNN
- Clustering
- Data Exploration & Dimensionality reduction
- Model Selection and Evaluation

k-Nearest Neighbor Classification (kNN)

- Unlike most of the learning methods, kNN does not build model from the training data.
- To classify a test instance *d*, define *k*-neighborhood *P* as *k* nearest neighbors of *d*
- Count number *n* of training instances in *P* that belong to class c_j
- Estimate $P(c_j|d)$ as n/k
- No training is needed. Classification time is linear in training set size for each test case.

Algorithm

1. Determine parameter K = number of nearest neighbors.
2. Calculate the distance between the query-instance and all the training samples.
3. Sort the distance and determine nearest neighbors based on the K-th minimum distance
4. Gather the category of the nearest neighbors
5. Use simple majority of the category of nearest neighbors as the prediction value of the query instance

- *k* is usually chosen empirically via a validation set or cross-validation by trying a range of *k* values.

- Distance function is crucial, but depends on applications.

Distance Metrics

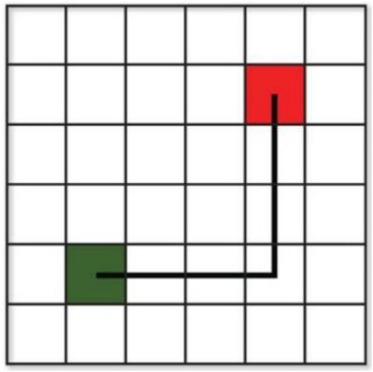
- Minkowski Distance
  - Non-negativity: d(x, y) >= 0
  - Identity: d(x, y) = 0 if and only if x == y
  - Symmetry: d(x, y) = d(y, x)
  - Triangle Inequality: d(x, y) + d(y, z) >= d(x, z)

$$\left(\sum_{i=1}^{n} |a_i - b_i|^p\right)^{1/p}$$

Distance Metrics

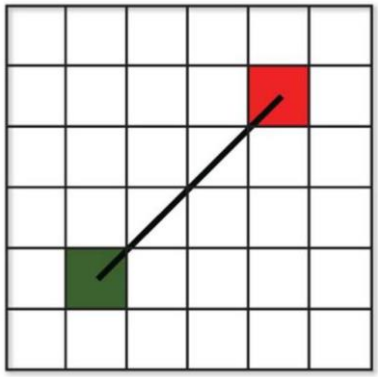
- Manhattan Distance

$$\sum_{i=1}^{n} |a_i - b_i|$$



- Euclidean Distance

$$\sqrt{\sum_{i=1}^{n} (a_i - b_i)^2}$$



Other distance metrics

- Cosine Distance

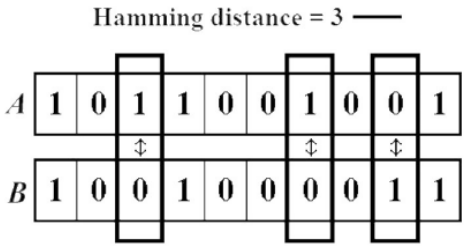
Calculate similarity between two vectors

$$1 - \cos\theta = 1 - \frac{\boldsymbol{a} \cdot \boldsymbol{b}}{\|\boldsymbol{a}\|\|\boldsymbol{b}\|}$$

- Jaccard Distance

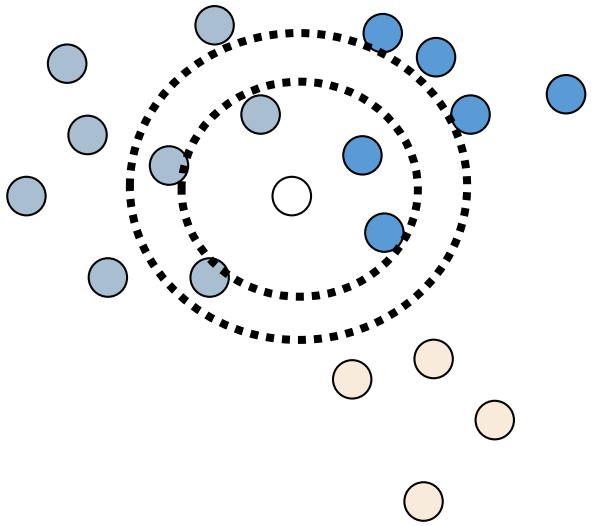
$$1 - J(A, B) = 1 - \frac{A \cap B}{A \cup B}$$

- Hamming Distance



Compare two binary data

Example



k=3 (3NN) ○ = ●

k=5 (5NN) ○ = ●

About KNN

- kNN can deal with complex and arbitrary decision boundaries.
- Despite its simplicity, researchers have shown that the classification accuracy of kNN can be quite strong and in many cases as accurate as those elaborated methods.
- kNN is **slow** at the classification time especially when the dataset is huge.
- kNN does not produce an understandable model

Questions?