K-NN and High Dimensional data

Instance-Based Classifiers

Set of Stored Cases

Atr1	………	AtrN	Class
			A
			B
			B
			C
			A
			C
			B

- Store the training records
- Use training records to predict the class label of unseen cases

Unseen Case

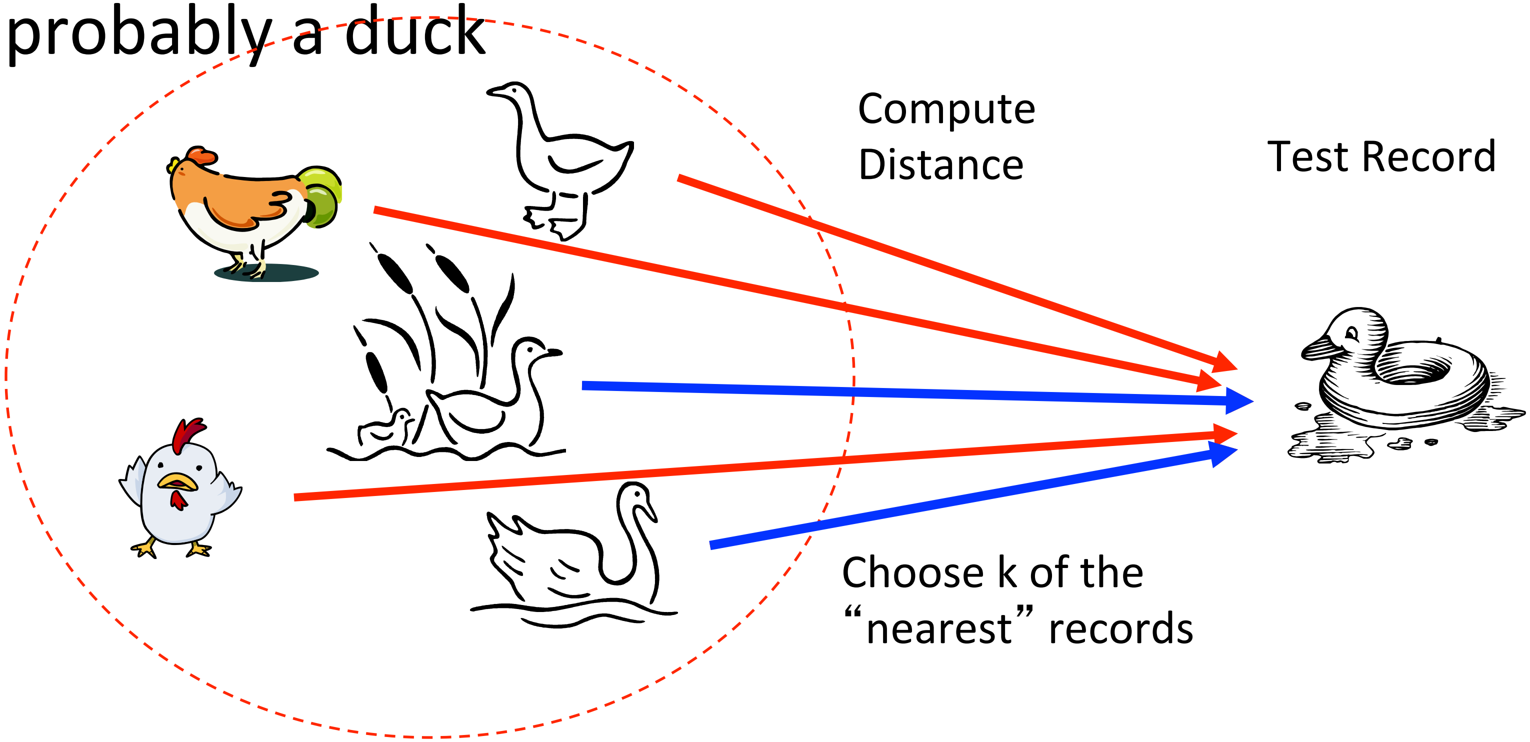
Atr1	………	AtrN

Instance Based Classifiers

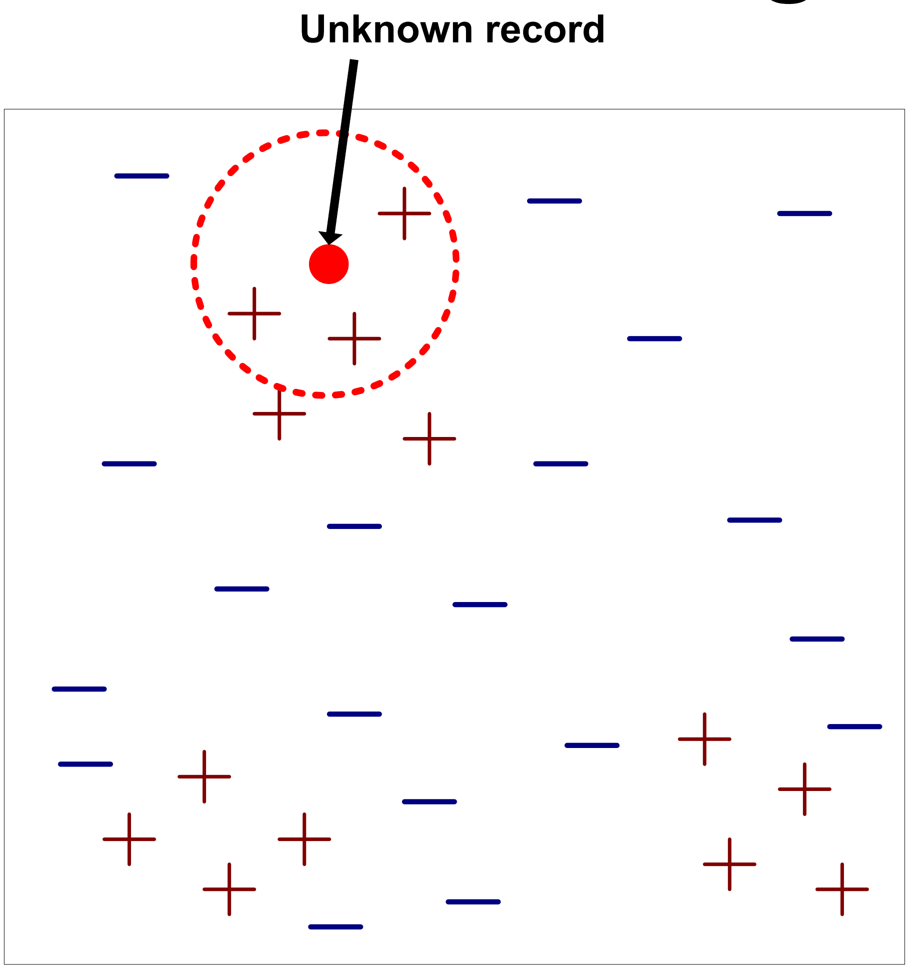
- Examples:
  - Rote-learner
    - Memorizes entire training data and performs classification only if attributes of record match one of the training examples exactly
  - Nearest neighbor
    - Uses k "closest" points (nearest neighbors) for performing classification

Nearest Neighbor Classifiers

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



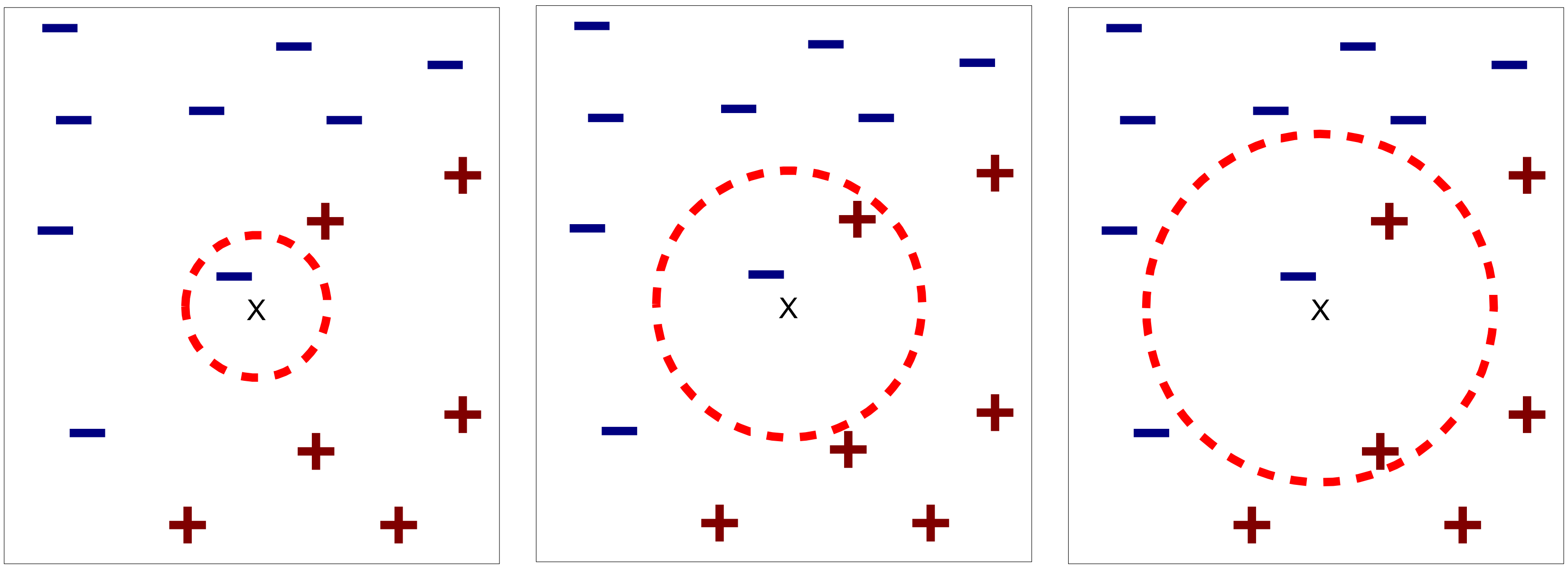
Nearest-Neighbor Classifiers



- Requires three things
  - The set of stored 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

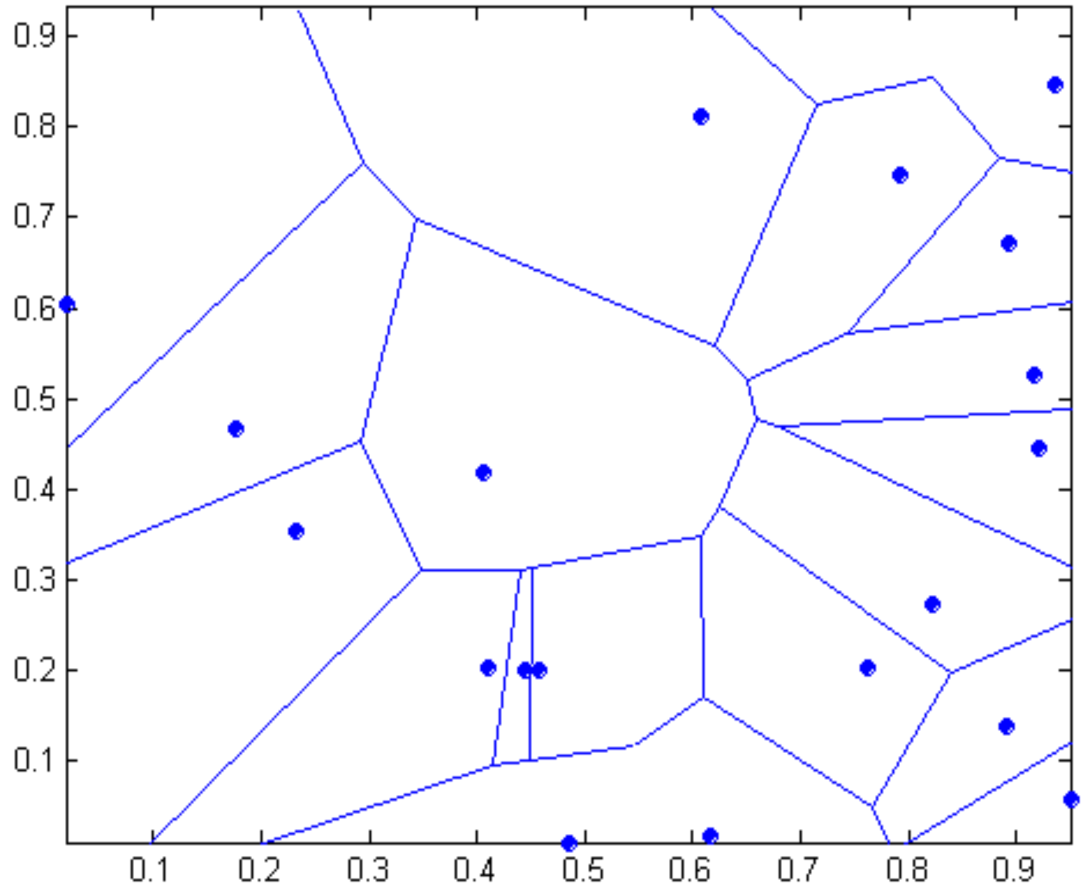


(a) 1-nearest neighbor (b) 2-nearest neighbor (c) 3-nearest neighbor

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

1 nearest-neighbor

Voronoi Diagram



Nearest Neighbor Classification

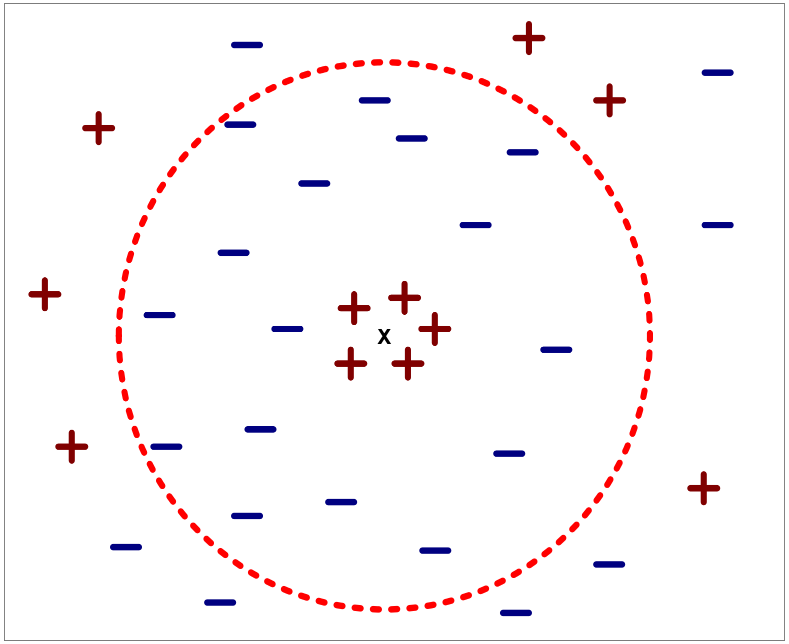
- Compute distance between two points:
  - Euclidean distance

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

- 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
  - If k is too large, neighborhood may include points from other classes



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 2.1m
    - 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

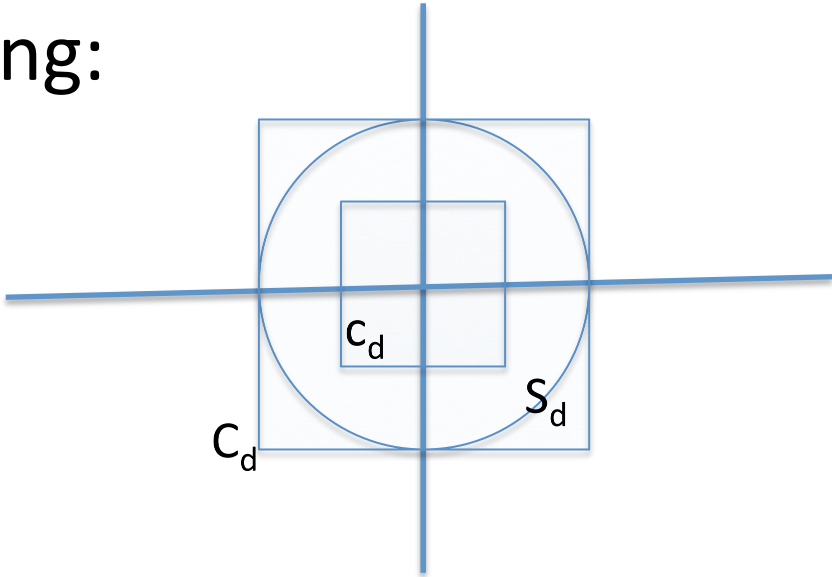
1 1 1 1 1 1 1 1 1 1 1 0	vs	1 0 0 0 0 0 0 0 0 0 0 0
0 1 1 1 1 1 1 1 1 1 1 1		0 0 0 0 0 0 0 0 0 0 0 1
d = 1.4142		d = 1.4142

High Dimensionality

- When data are in high dimensions, strange things happen….
- Consider the unit sphare in d-dimensions: S_d, the unit square c_d and the square C_d that contains completely the sphere and has length 2 in each dimension.
- In 2-d we have the following:

c_d is included completely

in S_d and S_d is inside C_d



High-d

- However, as d (dimensionality increases) let’s see what happens with the volumes of c_d, S_d, and C_d.

- $Vol(c_d) = 1$ as d-> ∞ (1^d)
- $Vol(C_d) = \infty$ as d-> ∞ (2^d)

But….. $Vol(S_d) = 0$ as d->∞!!!!!

Actually, in high dimensions, most of c_d lies outside S_d!!

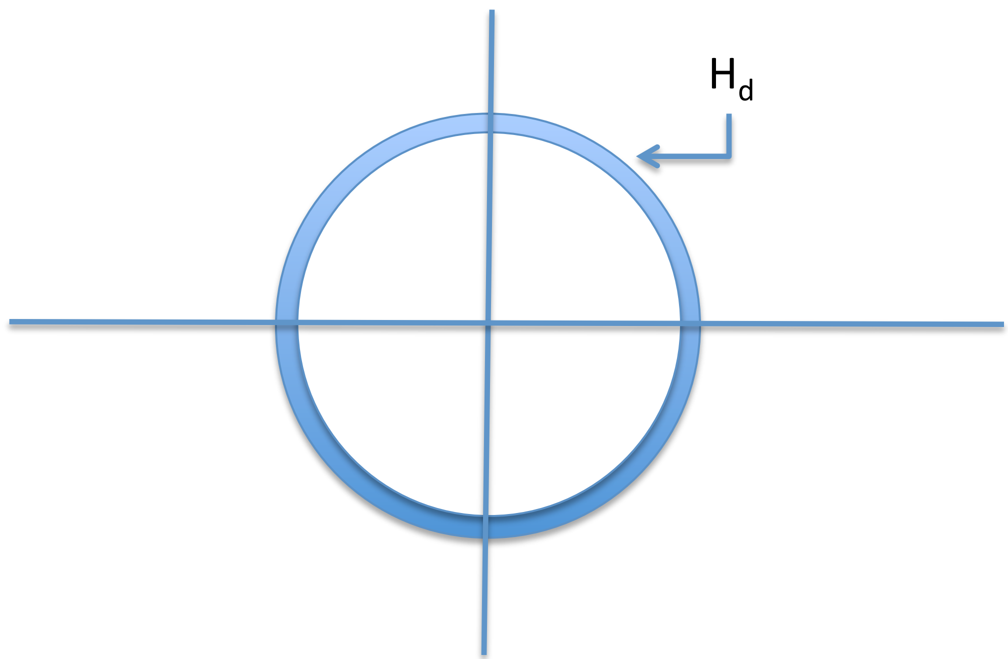
High-d

- Let's define the volume of S_d as V_d. Also, let consider the hyper-sphere E_d in d-dimensions with radius $1-\varepsilon$.
- Define the difference in volume between S_d and E_d as H_d. Then, $H_d = k_d\,(1^d - (1-e)^d)$
  k_d is a constant that depends on d.

Consider the ratio: $$\frac{H_d}{V_d} = \frac{k_d\,(1^d - (1-\varepsilon)^d)}{k_d\,1^d} \longrightarrow 1$$

that goes to 1 as d -> ∞

- Ratio goes to 1. So?....
- This means that the volume is concentrated on the shell (surface) around the surface of the hyper-sphere!!



High Dimensionality

- Hyper-sphere volume of unit radius goes to 0 as dimensionality goes to infinity!!!
- All data in the shell!!!
- In high dimensions, kNN can be problematic!

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