Applied Analytics and Predictive Modeling

Spring 2021

Lecture-12

Lydia Manikonda

manikl@rpi.edu

Rensselaer Polytechnic Institute · 1824 ·

Rensselaer

Some of the slides adapted from Intro to Data Mining Tan et al. 2nd edition

Today’s agenda

- Project details
- K-NN
- Weka demo – you can download from https://www.cs.waikato.ac.nz/ml/weka/

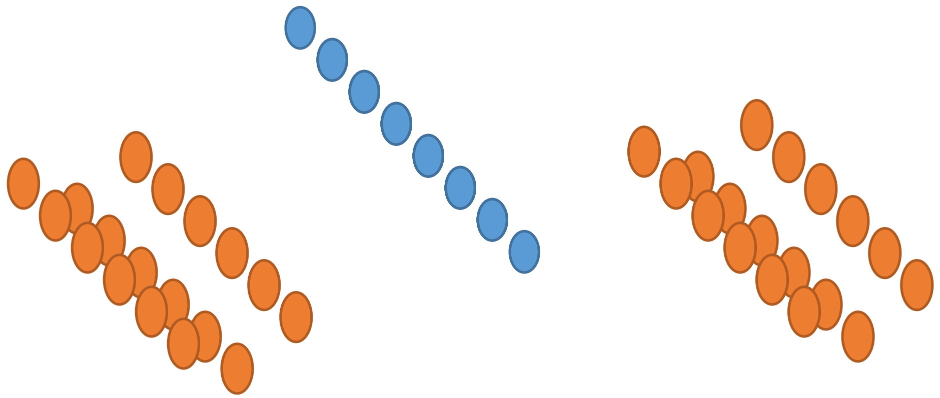
K-Nearest Neighbor Algorithm

Adapted from Intro to Data Mining, Tan et al., 2nd edition.

Background

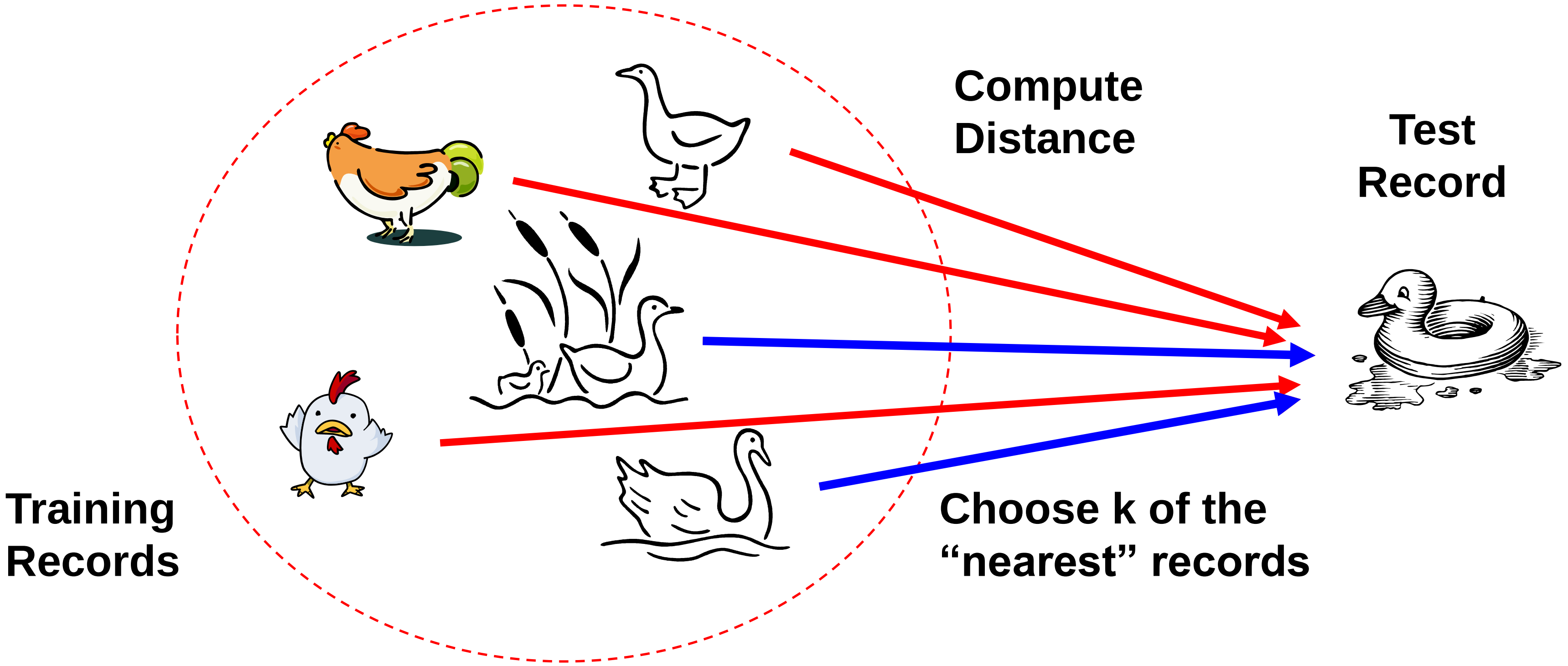
- Situations such as:
  - In Complex decision boundaries
  - If your data is coming in streams

- KNN can address these drawbacks

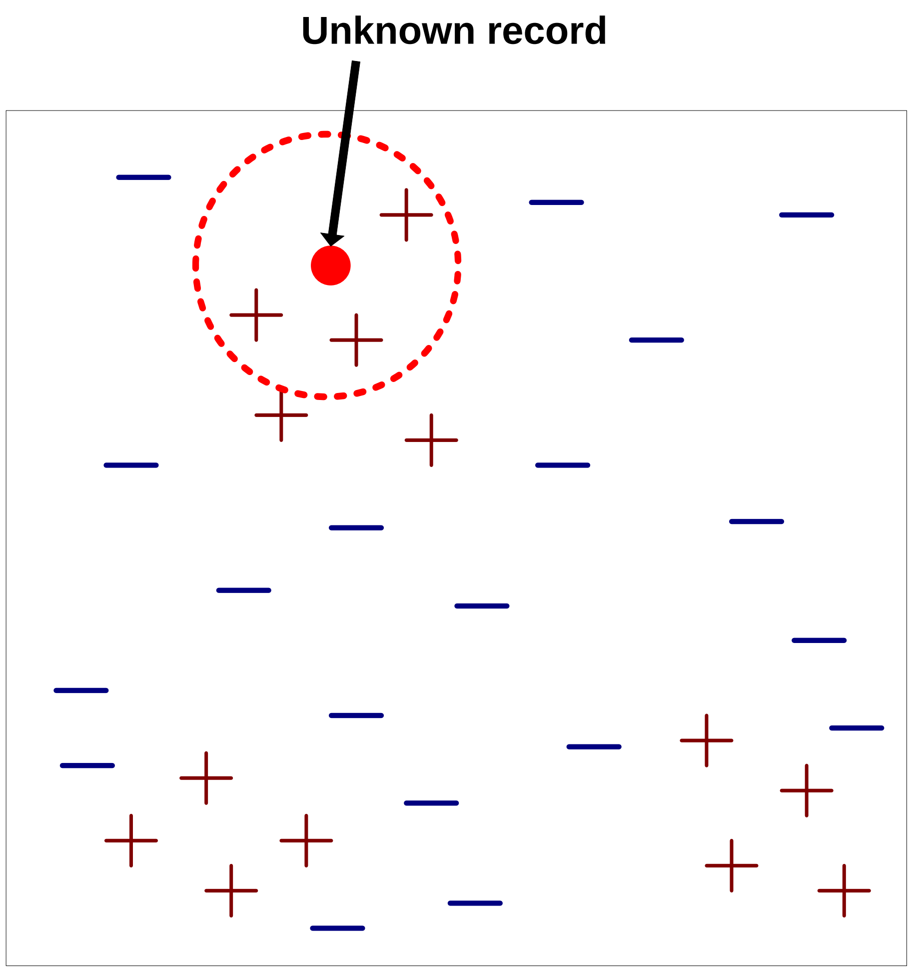


Nearest Neighbor Classifiers

- 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 labeled 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)

Nearest Neighbor Classification

- Compute proximity between two points:
  - Example: Euclidean distance

$$d(\boldsymbol{x}, \boldsymbol{y}) = \sqrt{\sum_i (\boldsymbol{x_i} - \boldsymbol{y_i})^2}$$

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

Example:

- Given this dataset, can you classify this sample data point using *K*-NN where *k*=3 and use Euclidean distance.

ID	Speed	Agility	Draft
1	400	6	Yes
2	71000	50000	Yes
3	100000	1	No
4	5000	7	Yes
5	1000	200000	No

To-do:

1. How many classes?
2. To which class does this data point (4,4) belong to?

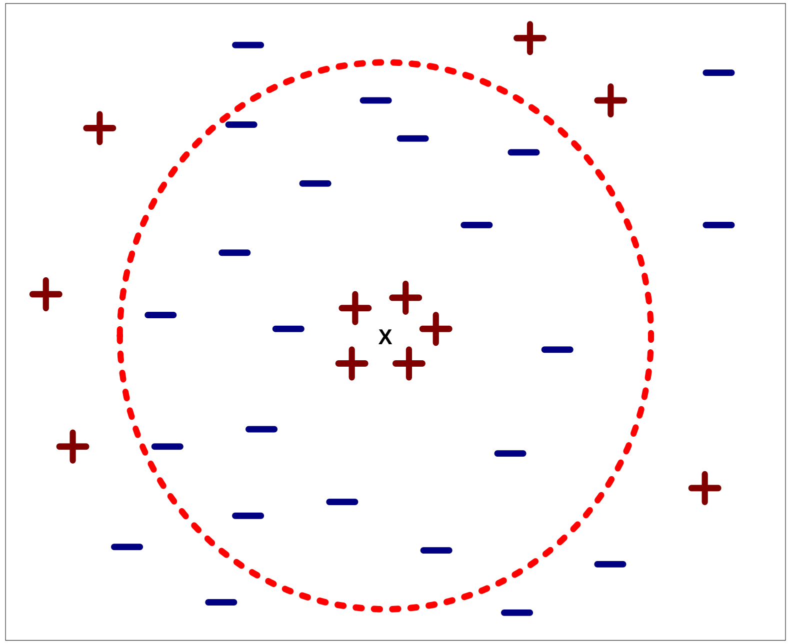
Can I use this dataset this way without any preprocessing or do I need to do preprocessing? If so, which operation and if not, why not?

- (4,4) -> (4,6) = sqrt(4) = 2 -- Yes
- → (7, 5) = sqrt(10) -- Yes
- → (1, 1) = sqrt(18) -- No
- → (5, 7) = sqrt((4-5)**2 + (4-7)**2) = sqrt(10) -- Yes
- → (1, 2) = sqrt(13) – No

- Closest-3 data points: {Yes, Yes, Yes} – Yes

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…

- **Choice of proximity measure matters**
  - For documents, cosine is better than correlation or Euclidean

1 1 1 1 1 1 1 1 1 1 1 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
1 0 0 0 0 0 0 0 0 0 0 0

Euclidean distance = 1.4142 for both pairs

Bag of words model

- Doc1 = “I love ice cream and its cold”
- Doc2 = “I love ice cream”

- Corpus = all the set of documents that you are considering.
- Vocabulary = {I, love, ice, cream, and, its, cold} = 7

- Doc1 = [1, 1, 1, 1, 1, 1, 1]
- Doc2 = [1, 1, 1, 1, 0, 0, 0]

- Doc1 = “my cat likes we fight”
- Doc2 = “my cat fight a lot fight fight”

- Vocabulary = {my, cat, likes, we, fight, a, lot} = 7
- Doc1 = [1, 1, 1, 1, 1, 0, 0]
- Doc2 = [1, 1, 0, 0, 3, 1, 1]

Nearest Neighbor Classification…

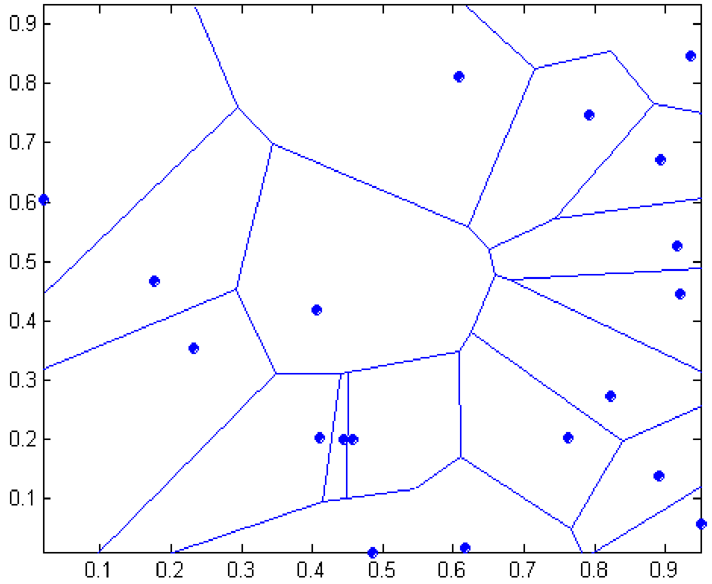
- **Data preprocessing is often required**
  - Attributes may have to be scaled to prevent distance measures from being dominated by one of the attributes
    - Example:
      - height of a person may vary from 1.5m to 1.8m
      - weight of a person may vary from 90lb to 300lb
      - income of a person may vary from $10K to $1M
  - Time series are often standardized to have 0 means a standard deviation of 1

Nearest-neighbor classifiers

- Nearest neighbor classifiers are local classifiers

- They can produce decision boundaries of arbitrary shapes.

1-nn decision boundary is a Voronoi Diagram



Nearest Neighbor Classification…

- **How to handle missing values in training and test sets?**
  - Proximity computations normally require the presence of all attributes
  - Some approaches use the subset of attributes present in two instances
    - This may not produce good results since it effectively uses different proximity measures for each pair of instances
    - Thus, proximities are not comparable

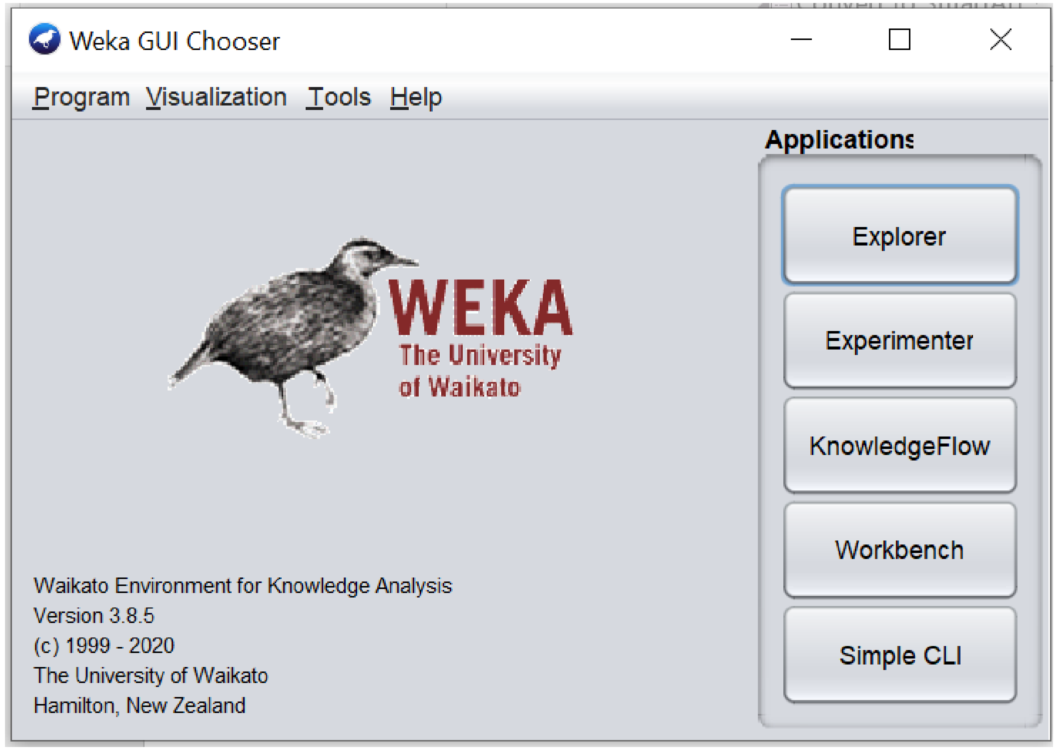
Nearest Neighbor Classification…

- **Handling irrelevant and redundant attributes**
  - Irrelevant attributes add noise to the proximity measure
  - Redundant attributes bias the proximity measure towards certain attributes
  - Can use variable selection or dimensionality reduction to address irrelevant and redundant attributes

Improving KNN Efficiency

- Avoid having to compute distance to all objects in the training set
  - Multi-dimensional access methods (k-d trees)
  - Fast approximate similarity search
  - Locality Sensitive Hashing (LSH)
- Condensing
  - Determine a smaller set of objects that give the same performance
- Editing
  - Remove objects to improve efficiency

Weka demo..



- Python notebook to follow..