

## Data Mining (Week 1)

# MSc Data Mining

## Topic 05 : Classification

### Part 04 : K Nearest Neighbours

Dr Bernard Butler and Dr Kieran Murphy

Department of Computing and Mathematics, WIT.  
(bernard.butler@wit.ie; kmurphy@wit.ie)





Spring Semester, 2022

## Outline

- K Nearest Neighbours (KNN) algorithm

# k-Nearest Neighbour Methods

## General Idea

- Given  $m$  labeled observations (, , and ) , how should we classify a new unlabelled observation ()?
- We could use the labels of the  $k$ -nearest neighbouring points.
  - “Nearest” means distance — how should we calculate this?
  - How do we pick the value for  $k$ ?
- What is our decision rule?
  - Assign new observation to most frequent occurring class in  $k$ -nearest neighbours.
  - What to do if there is a tie?

# Distance Functions

We frequently want to measure how close/near/similar two points (think observations/instances/cases) are. For this we need a distance function.

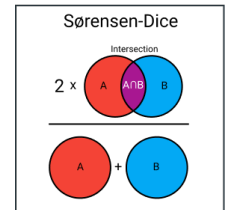
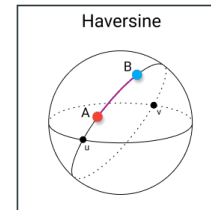
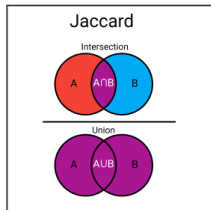
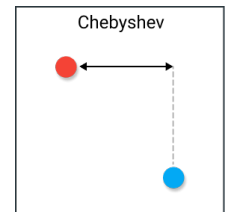
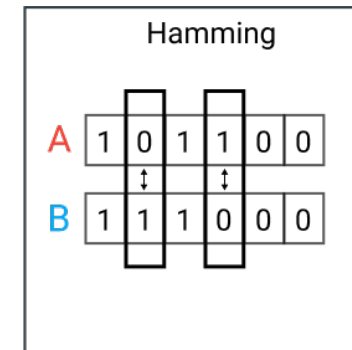
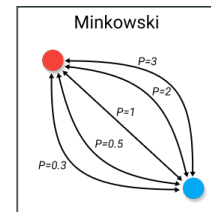
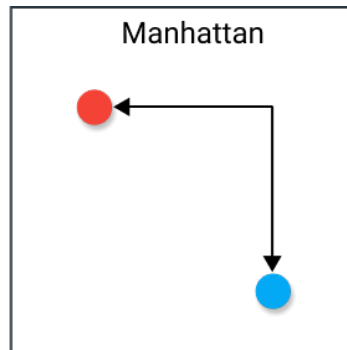
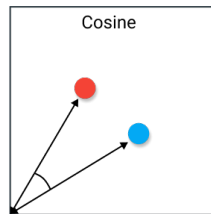
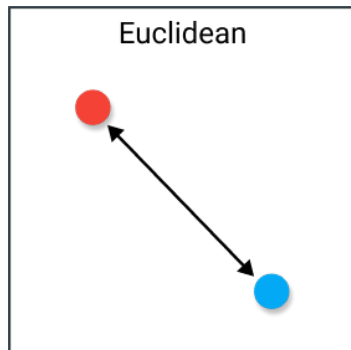
## Distance Function

A **distance function**,  $D(a, b)$ , is any function that satisfies the properties:

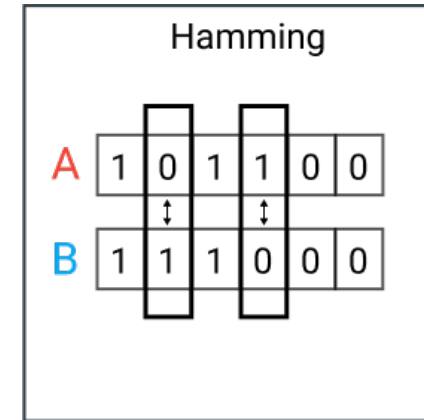
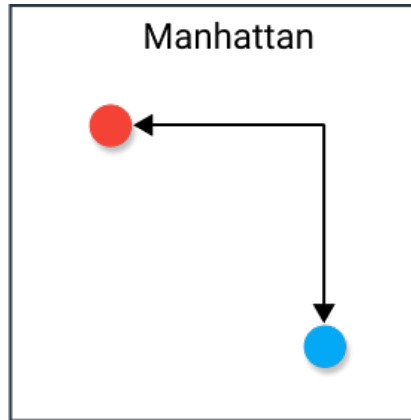
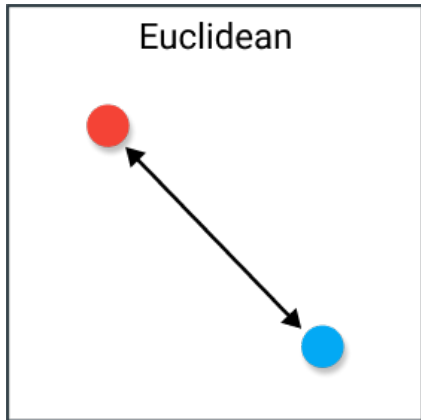
**non-negativity:**  $D(a, b) \geq 0$ , distance between any two points is non-negative and is only zero if  $a = b$ .

**symmetric:**  $D(a, b) = D(b, a)$

**triangular inequality:**  $D(a, c) \leq D(a, b) + D(b, c)$



# Distance Functions — Euclidean, Manhattan, and Hamming



- Pythagorean theorem

$$D(a, b) = \sqrt{\sum_{i=1}^n [a^{(i)} - b^{(i)}]^2}$$

- “As the crow flies”
- Features should be normalised before use
- ✓ Most commonly used metric.
- ✗ Becomes less useful for large dimensions

- Taxi-cab distance

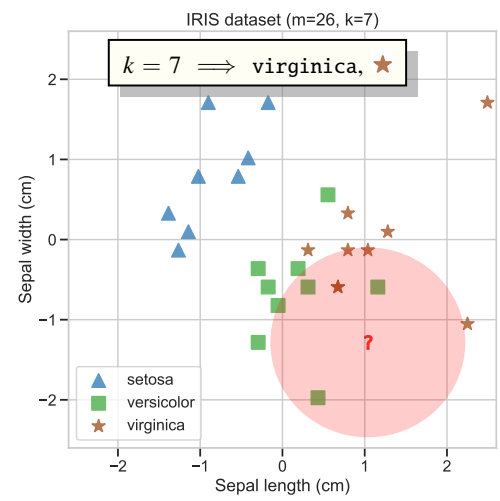
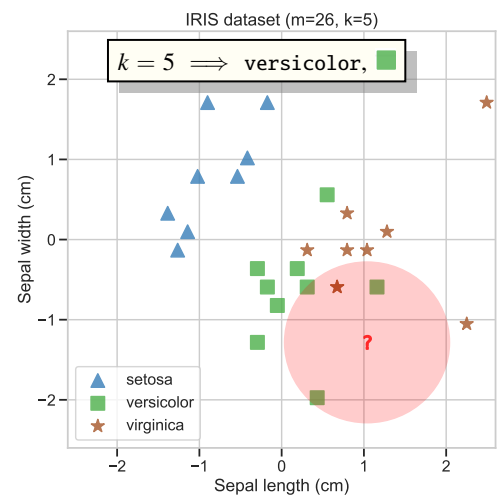
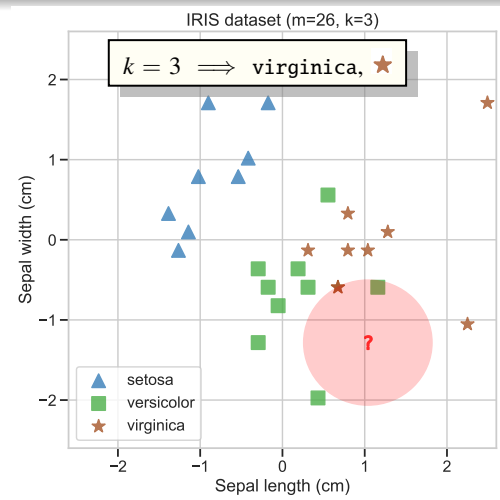
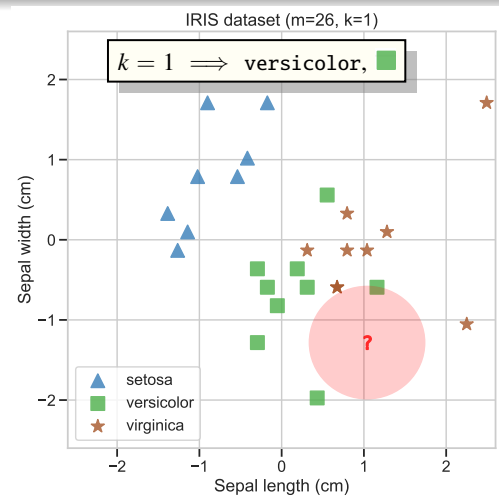
$$D(a, b) = \sum_{i=1}^n |a^{(i)} - b^{(i)}|$$

- ✓ Seems to work better than Euclidean for high-dimensional data
- ✓ Suitable for datasets with discrete and/or binary features.

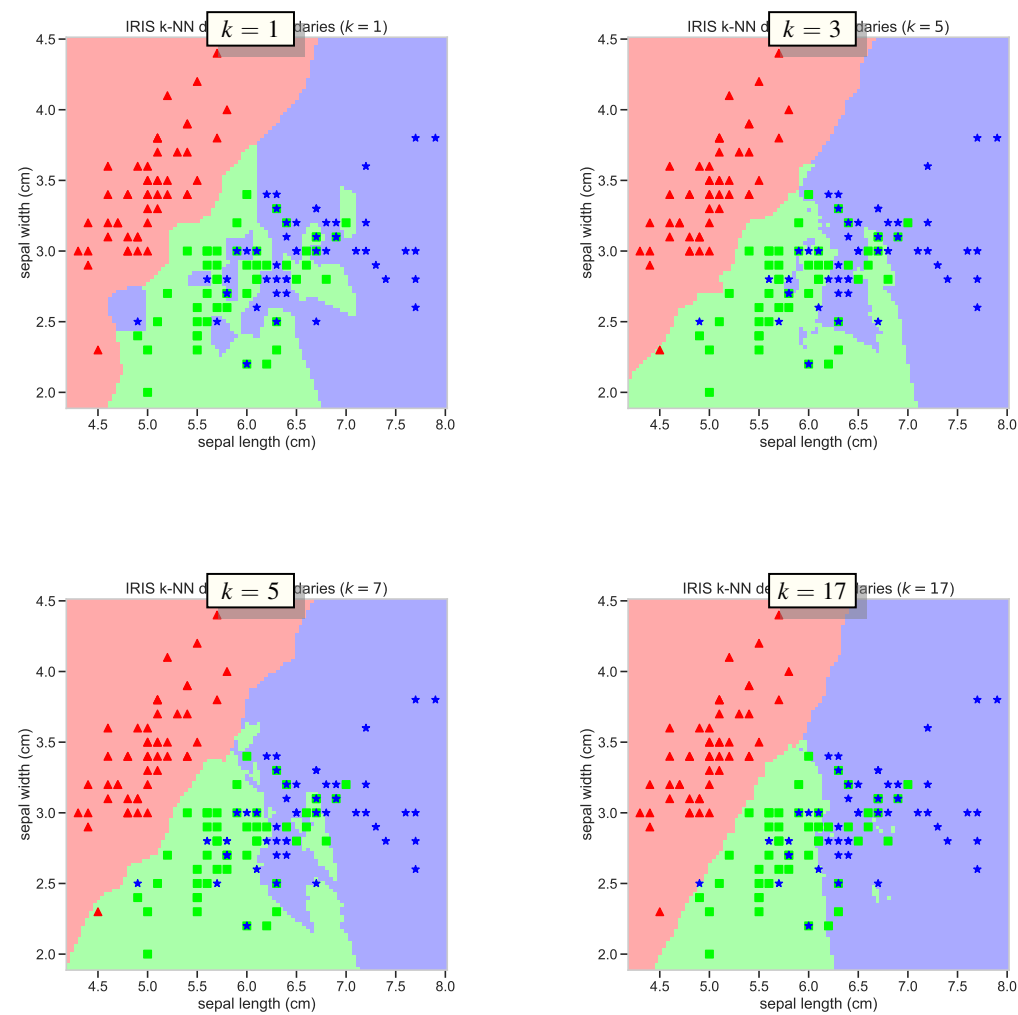
- Count of the number of differences (bits/letters/levels etc) between two points.

- ✓ Can be used between categorical variables.
- ✗ Difficult to use when two vectors are not of equal length.
- ✗ Should not be used when magnitude is important.

# Effect of $k$



# Effect of $k$ on Decision Boundary



# k-Nearest Neighbour Methods — Review

## When to Consider

- Observations/instances map to points in  $\mathbb{R}^n$  (quantitative/numerical features)
- Less than 20 features/attributes per instance (low dimensionality)
- Lots of training data (more points means closer neighbours)

## Advantages

- Training is very fast (instantaneous, since lazy learner)
- Learn complex target functions
- Do not lose information (lazy learner)

## Disadvantages

- Slow at query time (uses training data not model to predict)
- Memory-based technique (must pass over (nearly) all points for each classification)
- Easily fooled by irrelevant features/attributes

## Hyper-Parameters

- Distance metric (Euclidean — “as the crow flies”)
- Number of neighbours,  $k$  (Increasing  $k$  reduces variance, increases bias)

# Resources

---

- 9 Distance Measures in Data Science

[towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa](https://towardsdatascience.com/9-distance-measures-in-data-science-918109d069fa)

Non-technical comparison of common distances functions (source of images used here).