Algorithms and their Applications CS2004 (2020-2021)

Dr Mahir Arzoky

17.1 Bin Packing and Data Clustering

CodeRunner Examination (Task #2) Assessment Brief was released!

Coursework and CodeRunner…

- **Coursework (60%)**
  - Task #1 (CodeRunner Class Tests)
    - 30%
    - Already completed
  - Task #2 (CodeRunner Examination)
    - 70%
    - Will be held in Week 25
    - During your scheduled laboratory sessions

Exam

- Exam (40% weight)
  - Timed (3 hours), online and open-book
  - WiseFlow and held during the University’s May examination period
  - Theory based
  - There will be NO programming needed in the exam
  - Past exam papers are already on Blackboard!
  - But, the format this year is different!
    - No multiple choice questions!

Previously On CS2004…

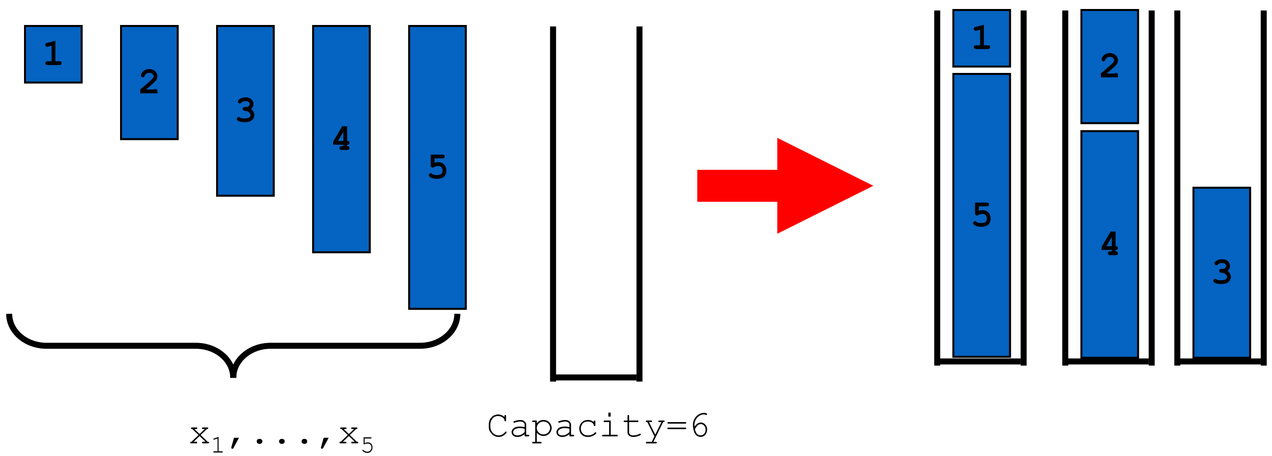
- So far we have looked at:
  - Concepts of Computation and Algorithms
  - Comparing algorithms
  - Some mathematical foundation
  - The Big-Oh notation
  - Computational Complexity
  - Data structures
  - Sorting Algorithms
  - Various graph traversal algorithms
  - Heuristic Search
  - Hill Climbing and Simulated Annealing
  - Parameter Optimisation (Applications)
  - Evolutionary Computation
  - Swarm Intelligence
  - Travelling Salesperson Problem

This Lecture

- Within this lecture we are going to look further at a number of algorithms
- We will look at:
  - Bin packing (briefly)
  - Data Clustering (in a bit more detail)

Bin Packing

- The **bin packing** problem is where a number of n items of size $x_1, \ldots, x_n$ need putting into the smallest number of bins (or boxes) of size/capacity c



Bin Packing Algorithms

- Combinatorial problem
- There are a large number of bin packing applications:
  - Filing recycle bins / loading trucks
  - CD/tape compilations
  - TV/radio advertisements
  - Cutting stock
- There are a large number of bin packing methods
- We will look at the **first-fit decreasing** bin packing algorithm

First-Fit Decreasing (FFD)

- Anyone who has tried packing a suit case knows that you pack the biggest items first and leave the smallest items to last!
- This algorithm takes advantage of this idea
  - n empty bins are created and numbered 1.. n
  - The items that need to be packed are **sorted** in decreasing order
  - Each item is packed into the first bin it will fit into, starting at the largest first
  - Empty bins (on completion) are discarded/ignored
- The complexity is O(nlog(n)) plus the sorting algorithm used

Data Clustering – Part 1

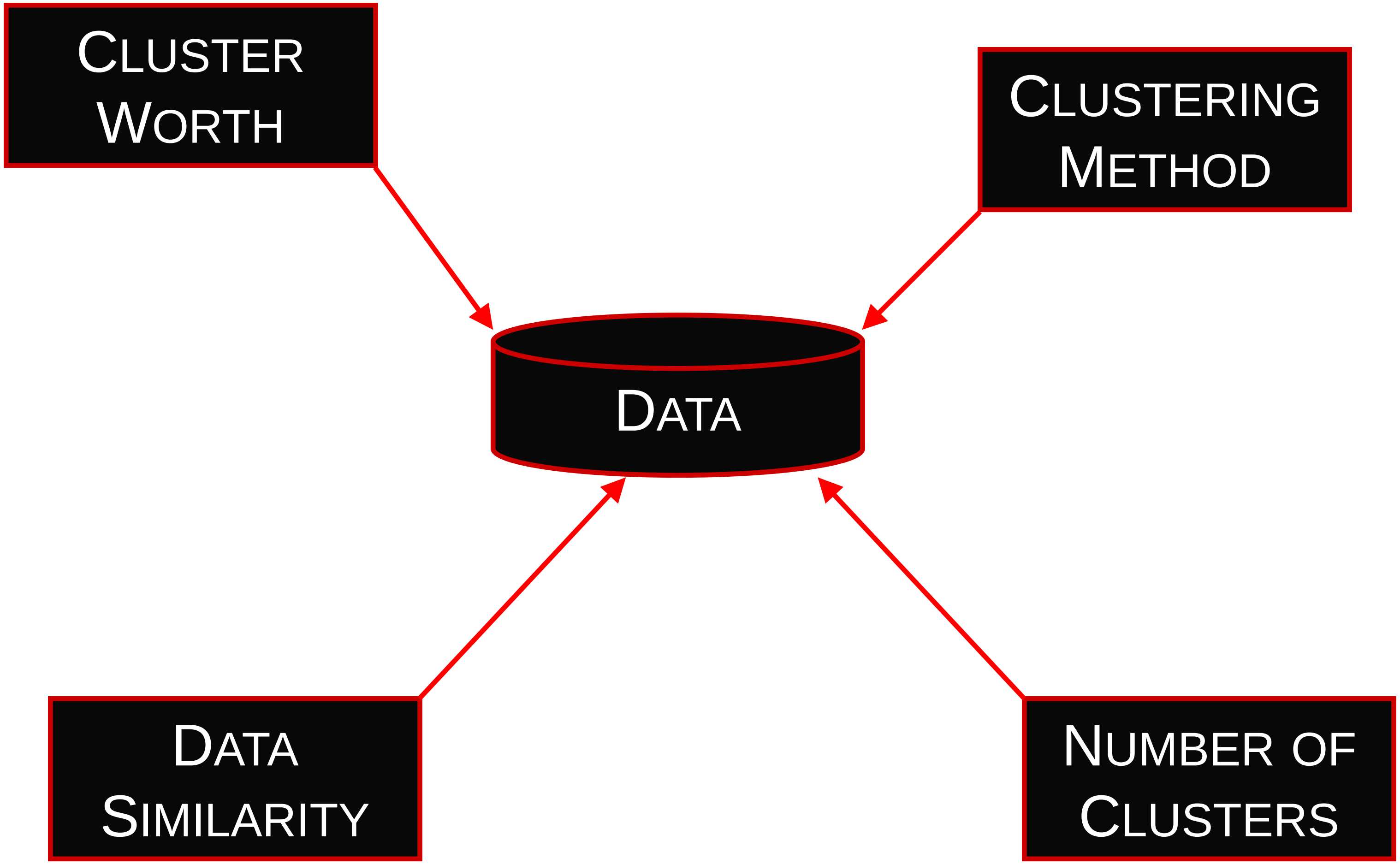
- **Data Clustering** is a common technique for data analysis
  - Used in many fields e.g. machine learning, pattern recognition, image analysis and bioinformatics, etc..
- **Data Clustering** is the process of arranging objects (as points) into a number of sets (k) according to “distance”
  - Each set (ideally) shares some common trait - often similarity or proximity for some defined distance measure
  - Each set will be referred to as a cluster/group
  - For the purposes of this module, each set is mutually exclusive, i.e. an item cannot be in more than one cluster

Data Clustering – Part 2

- The data that we are clustering usually consists of a number of examples (rows) (n) where we have measured a number of features (variables) (m =3 in the example below)
- We want to cluster the rows together based on how similar their features are
- We shall assume that the data we are clustering is a table or matrix X, where $\underline{x}_i$ is the ith row of X and x_{ij} is the jth variable (feature) of row i
  - For example x_{92} is 2.9 in the table below:

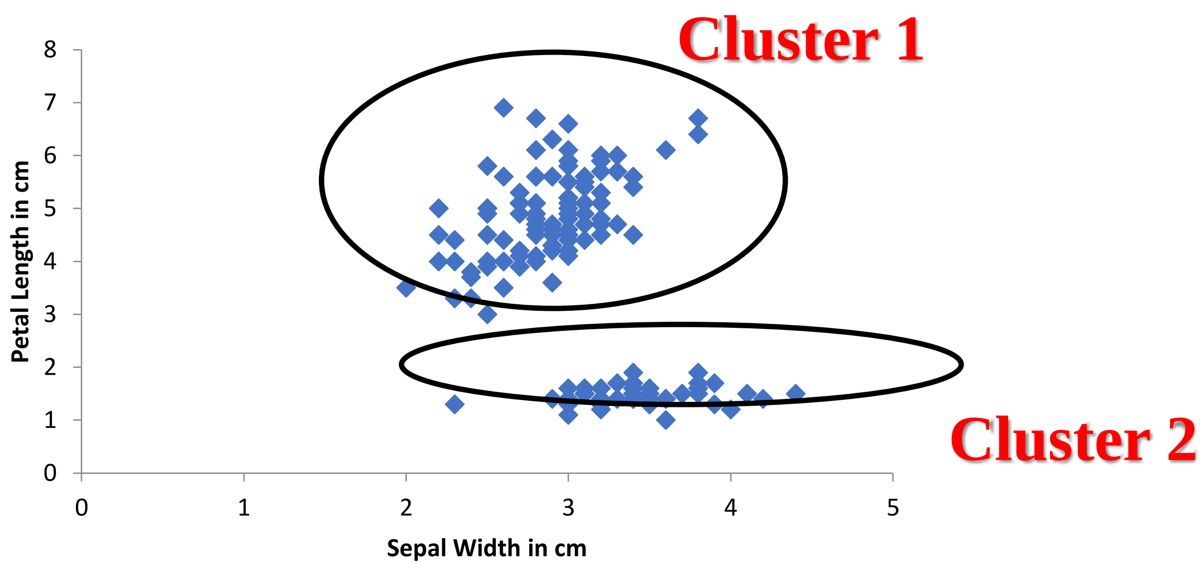
Sample	Sepal Length in cm	Sepal Width in cm	Petal Length in cm
1	5.1	3.5	1.4
2	4.9	3.0	1.4
3	4.7	3.2	1.3
4	4.6	3.1	1.5
5	5.0	3.6	1.4
6	5.4	3.9	1.7
7	4.6	3.4	1.4
8	5.0	3.4	1.5
9	4.4	**2.9**	1.4
10	4.9	3.1	1.5
Etc…	5.4	3.7	1.5

Data Clustering – Part 3



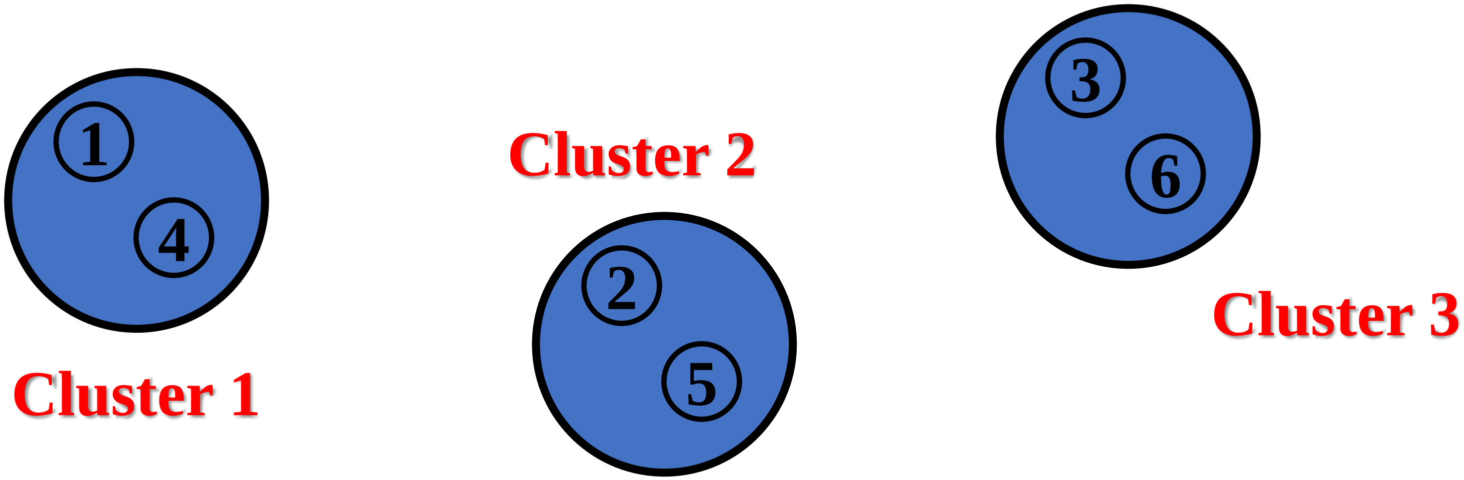
Data Clustering – Part 4

- If we are only clustering on two features or variables (m=2) then we can often plot the data and the clusters can be visualised
- However if we have hundreds of features....



Representing a Cluster

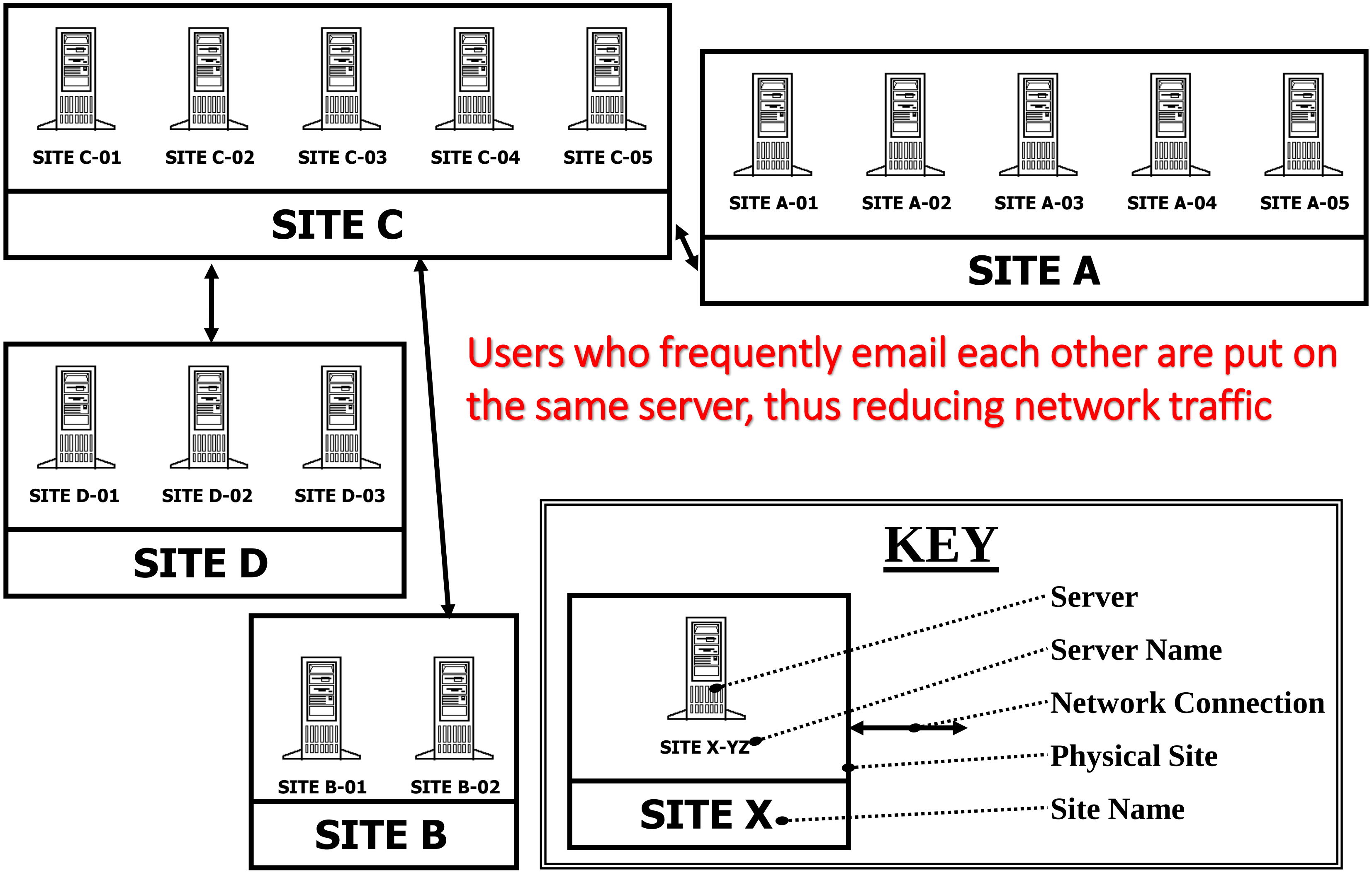
- A cluster will be represented as a vector C where $c_i = j$ means that object/item/row i is in cluster j
- For example C = {1,2,3,1,2,3} (k=3)



Why Cluster?

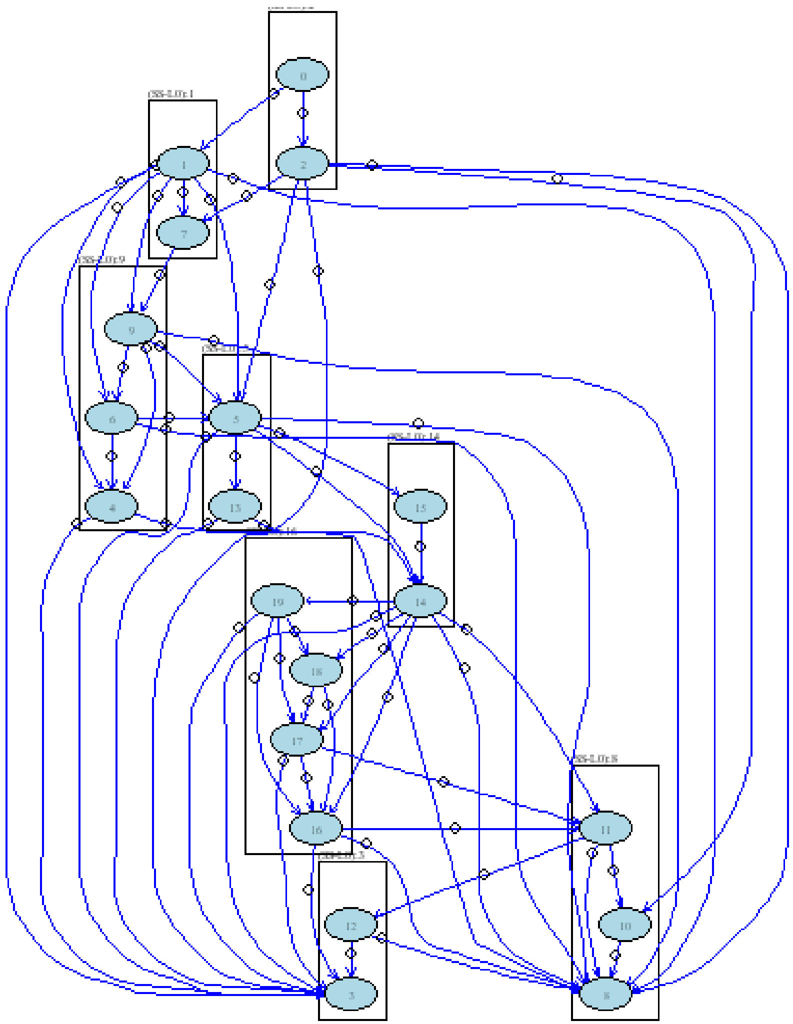
- Knowing which objects are highly related to other objects is very useful within data analysis
  - Less complex to model
  - A useful pre-processing tool
  - May give insight into the unknown properties of some of the objects

Application – Email Logfiles



Users who frequently email each other are put on the same server, thus reducing network traffic

Application – Modularisation



- ❑ Arrange "Software Components" into related modules
- ❑ Based on a binary relationship matrix

Data Similarity – Part 1

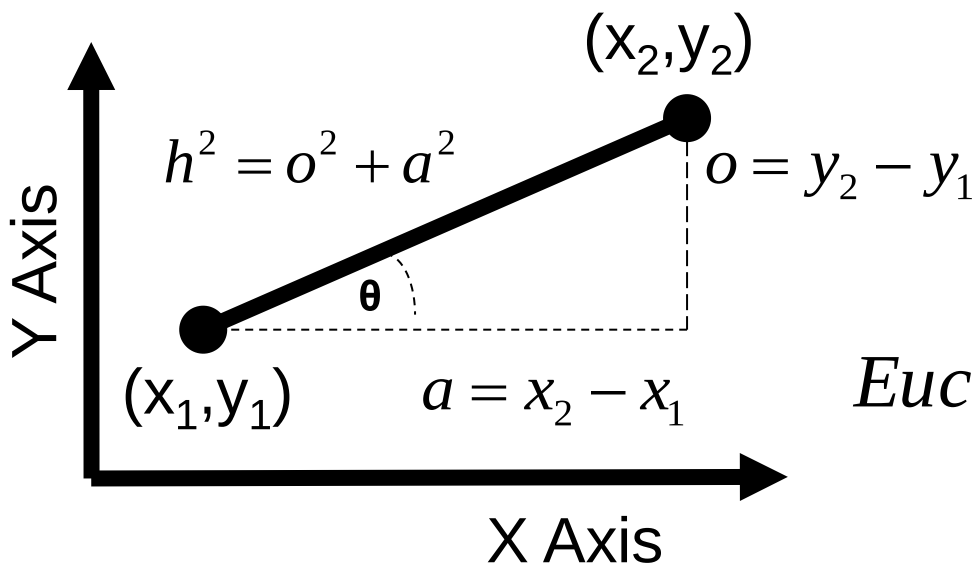
- Many methods are designed to work on **Distance Metrics** or **Similarity** between rows
  - E.g. K-Means
- Rows are compared to each other and a measure of how similar they are is used by the clustering methods
- Similar rows are placed into the same cluster

Data Similarity – Part 2

- There are many was to measure similarity between the objects that we are clustering
  - Euclidean
  - Correlation
    - Pearson
    - Spearman
    - Kendal
  - Manhattan
  - Etc...

Euclidean Distance

- The shortest distance between two points
- In the two dimensional case, this is the length of the hypotenuse of the right angled triangle constructed between two points (**Pythagoras's Theorem**)
- The **Euclidean** distance between two n-dimensional points or two data objects stored as a row vector is defined as follows:



$$Euclid(\underline{x}, \underline{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

Cluster Worth

- The choice of correct metric for judging the worth of a clustering arrangement is vital for success
  - E.g., Dense close clusters? Sparse far clusters?
- There are as many metrics as methods!
  - Sum of squares by cluster
  - Homogeneity (H) i.e. Density of clusters
  - Separation (S) i.e. Distance between clusters
  - H/S
  - Maximum likelihood
  - Etc…

Cluster Worth – Sum of Squares

- **K-Means** clustering (which we will look at later) judges the worth of a clustering arrangement based on the square of how far each item in the cluster is from the centre
- This is the sum of squared Euclidean distances
- C is a cluster of size k, $\underline{x}_i$ an element in the cluster and $\underline{c}$ is the centre of the cluster

$$SS(C) = \sum_{i=1}^{k} \left(Euclid(\underline{x}_i, \underline{c}) \right)^2$$

The Number of Clusters

- Many applications specify the number of clusters a solution requires, e.g. the email server application
- Many do not, e.g. gene expression data
- Determining the number of clusters is very difficult
- A choice of method that locates the number of clusters and their contents is often desirable

Methods

- Many different clustering approaches and algorithms
- Centroid-based clustering
  - K-Means
- Hierarchical clustering
- Density-based clustering
- Distribution-based clustering

K-Means Clustering

- This method requires the number of clusters (k) to be known
- The algorithm works by maintaining k cluster means called **centres**
- Objects (rows) are assigned to the closest centre and then the means are updated
- The algorithm terminates when the centres do not change or a fixed number of iterations has been conducted

The K-Means Algorithm

```
Algorithm 1. KMeans(X,k)
Input: Dataset X
       Required number of clusters k
1) Assign the objects (rows) randomly to k
   clusters ensuring no cluster is empty (c_1,…,c_k)
2) Calculate the centres of each cluster
3) Allocate each object to the new
   centres by minimising the sum of
   squares error, SS(c_i)
4) Repeat steps 2 and 3 until the
   terminating condition is met
Output: Set of clusters
```

How Good is a Clustering Arrangement?

- Once data is clustered, a data analyst would want to know if the results are any good!
  - Did they select the correct method?
  - Did they select the correct way of comparing objects/rows (distance metric)?
  - Do the results agree with what is known about the dataset?
  - Are the results consistent?
- Due to difficulty of problem, no direct way of addressing these questions
- Few ways to obtain insight into how the cluster method performed
  - Cluster worth
  - Expert knowledge
  - Comparing clusters

Comparing Clusters and Kappa Metric

- Metrics exist to measure how similar two clustering arrangements are
- Thus if a method produces a set of similar clustering arrangements (according to the metric) then the method is consistent
- We will consider the **Kappa** metric which has been adapted from Medical Statistics
- Kappa is an agreement metric defined for the comparison of two clustering arrangements

Kappa

Kappa	Agreement Strength
$-1.0 \le \kappa \le 0.0$	VERY POOR
$0.0 < \kappa \le 0.2$	POOR
$0.2 < \kappa \le 0.4$	FAIR
$0.4 < \kappa \le 0.6$	MODERATE
$0.6 < \kappa \le 0.8$	GOOD
$0.8 < \kappa \le 1.0$	VERY GOOD

The Kappa Guideline

Next Lecture

- There is no lecture next week!
  - Only one lecture remaining (revision lecture in Week 30)…
  - Details will be posted on Blackboard…

Next Laboratory

- ❑ The laboratory will involve running and comparing K-Means clustering
- ❑ The laboratory sessions will continue next week!