Approximate Hierarchical Clustering of Large Datasets

Meelis Kull

Supervisor: Dr. Jaak Vilo

Pedase,

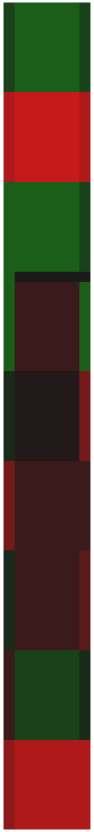
October 3, 2003

Overview

- Gene expression data
- What is clustering?
- What is hierarchical clustering?
- Why need for speedup?
- Approximate hierarchical clustering
- Finding closest pairs of data items fast
- Results
- Problems
- Future

Gene expression data (1)

Sample

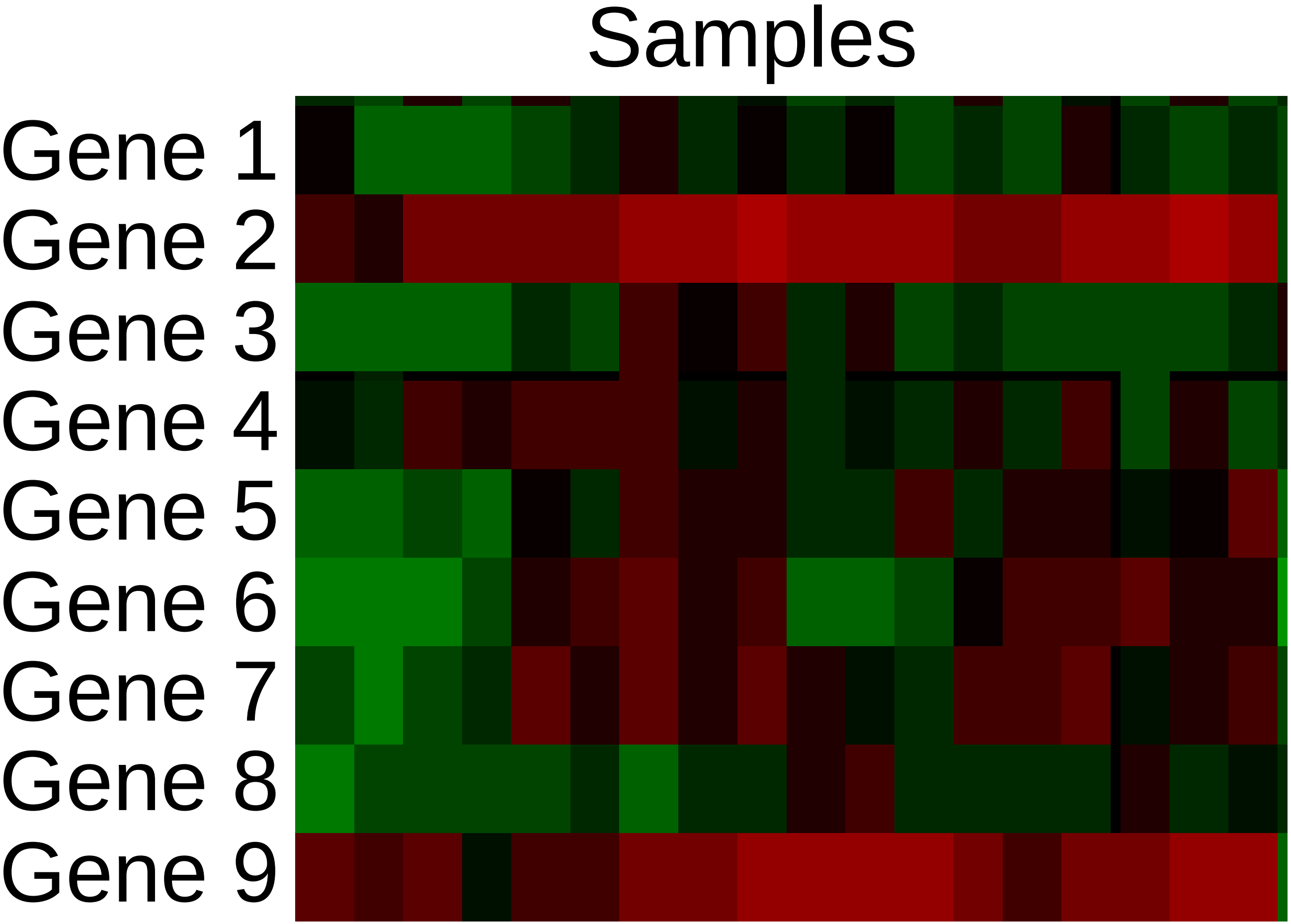


Gene 1	-2.303
Gene 2	+2.904
Gene 3	-2.235
Gene 4	+0.572
Gene 5	-1.169
Gene 6	+0.824
Gene 7	+0.343
Gene 8	-1.678
Gene 9	+2.477

🟥 Gene is highly expressed

🟩 Gene is not expressed

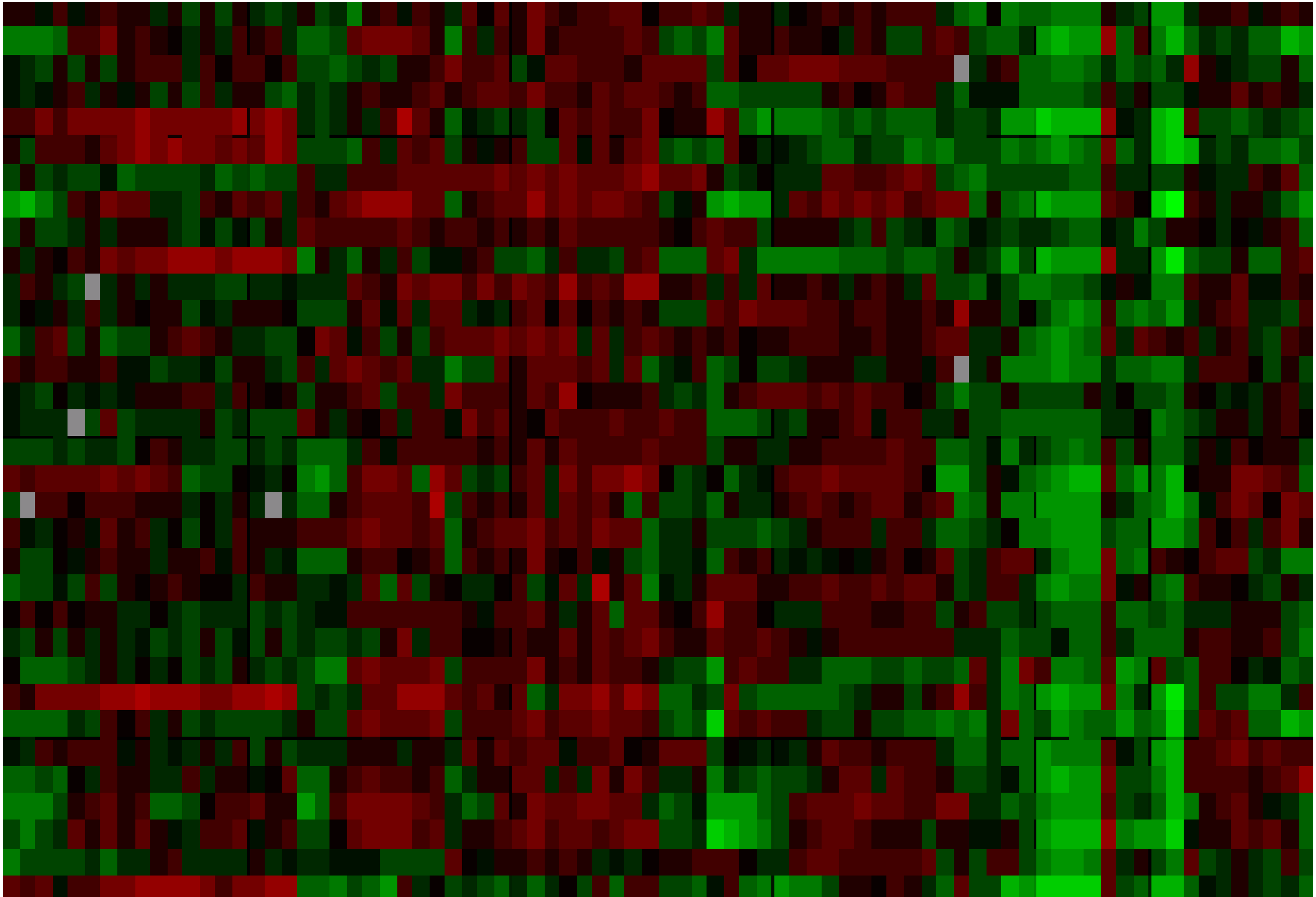
Gene expression data (1)



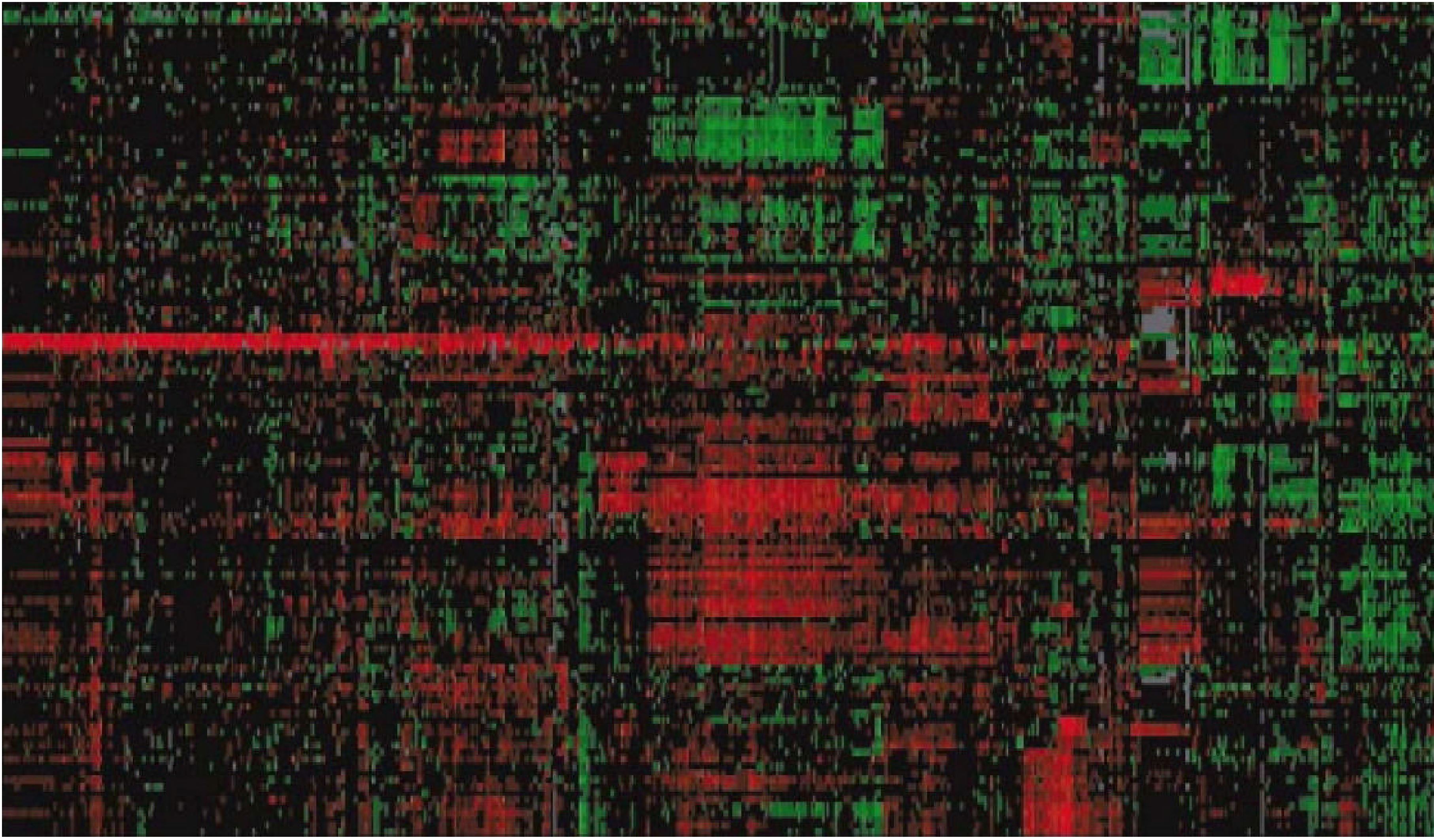
🟥 Gene is highly expressed

🟩 Gene is not expressed

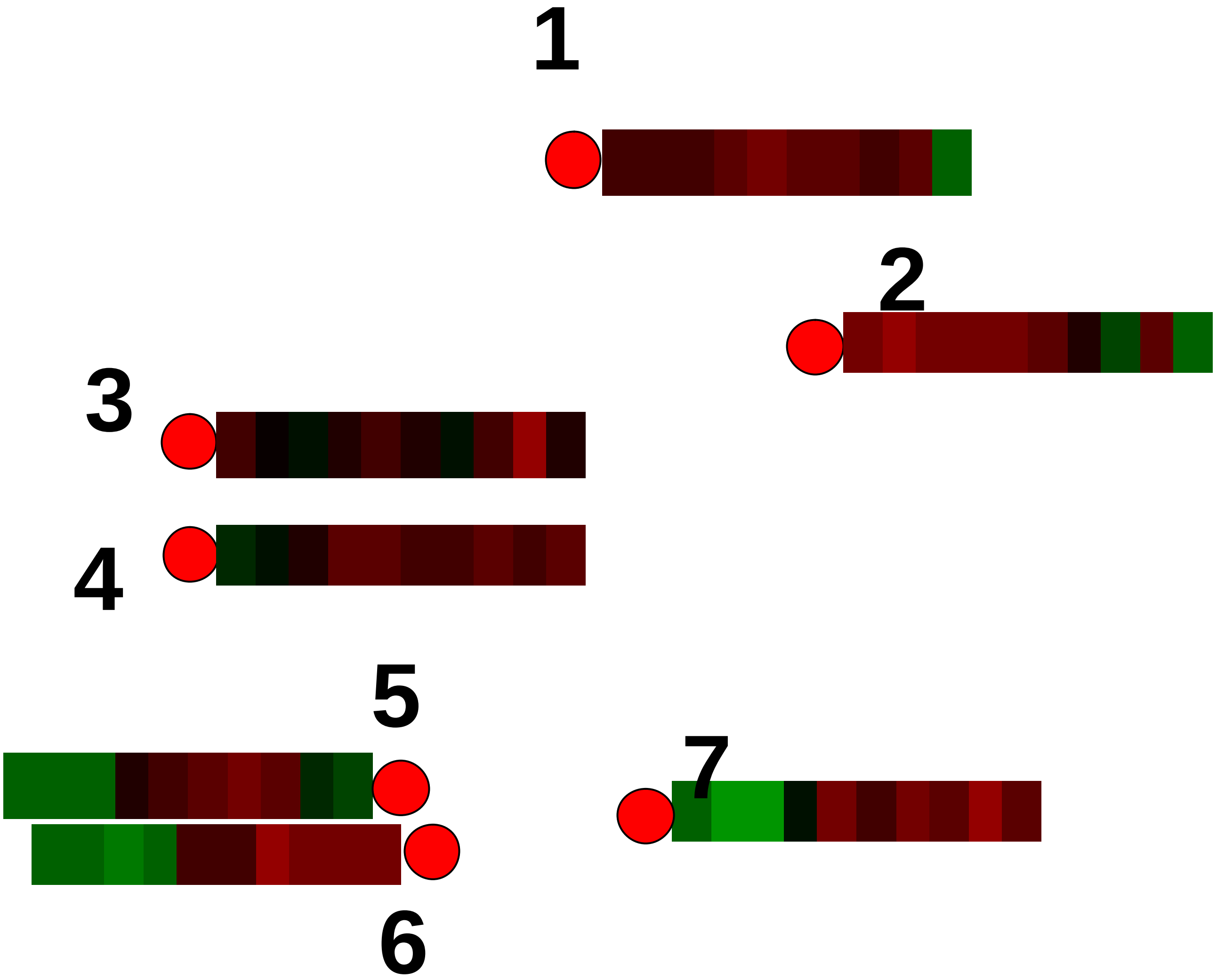
Gene expression data (2)



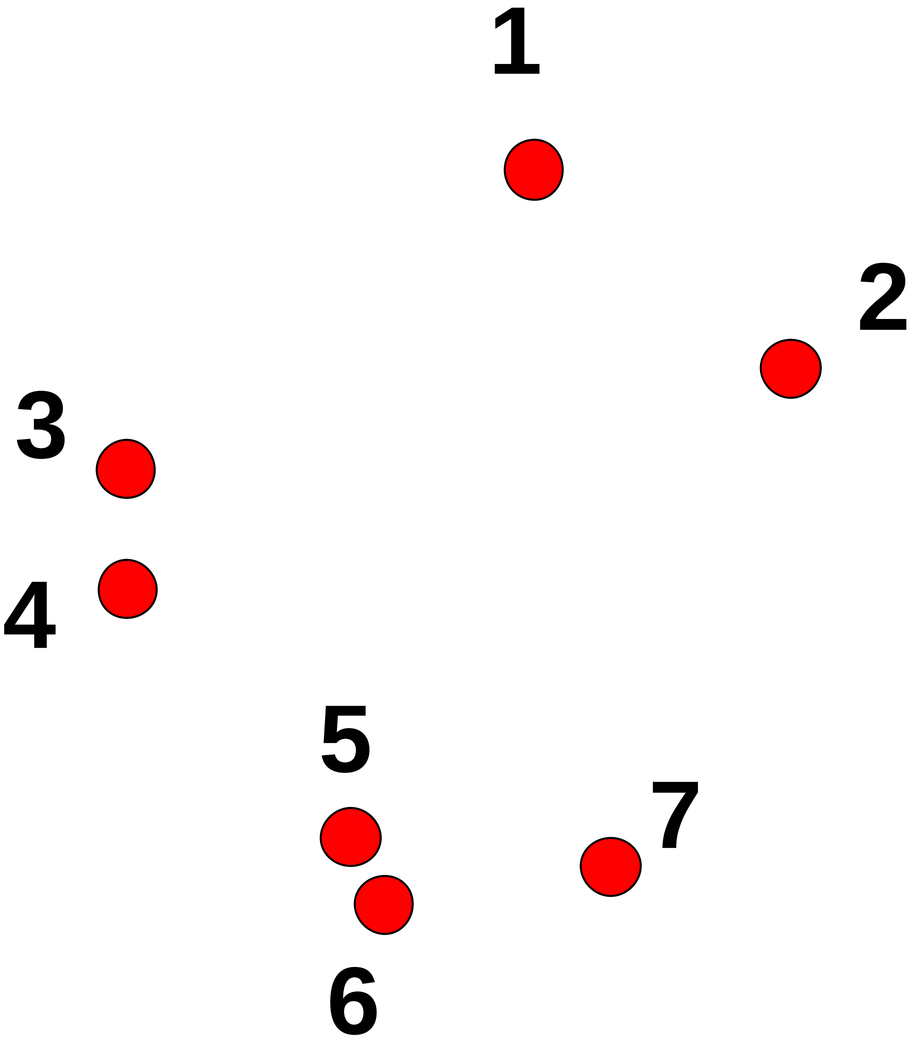
Gene expression data (3)



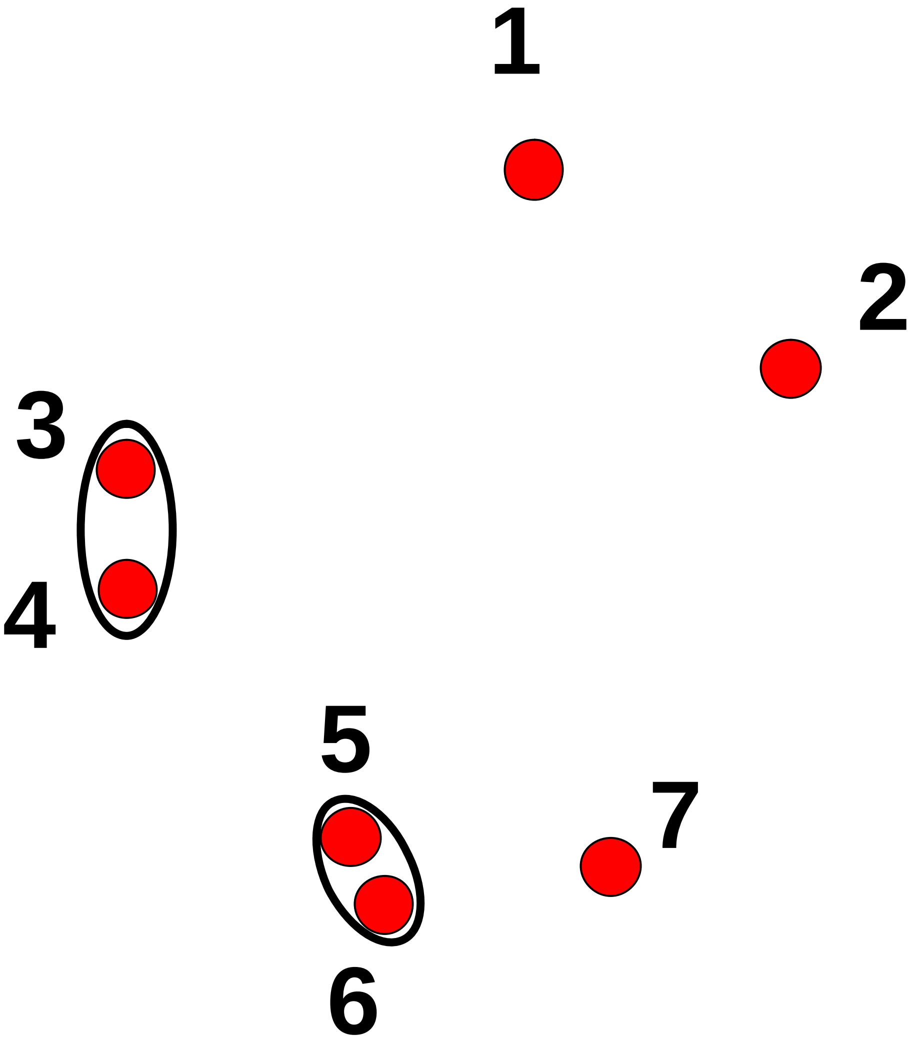
What is clustering?



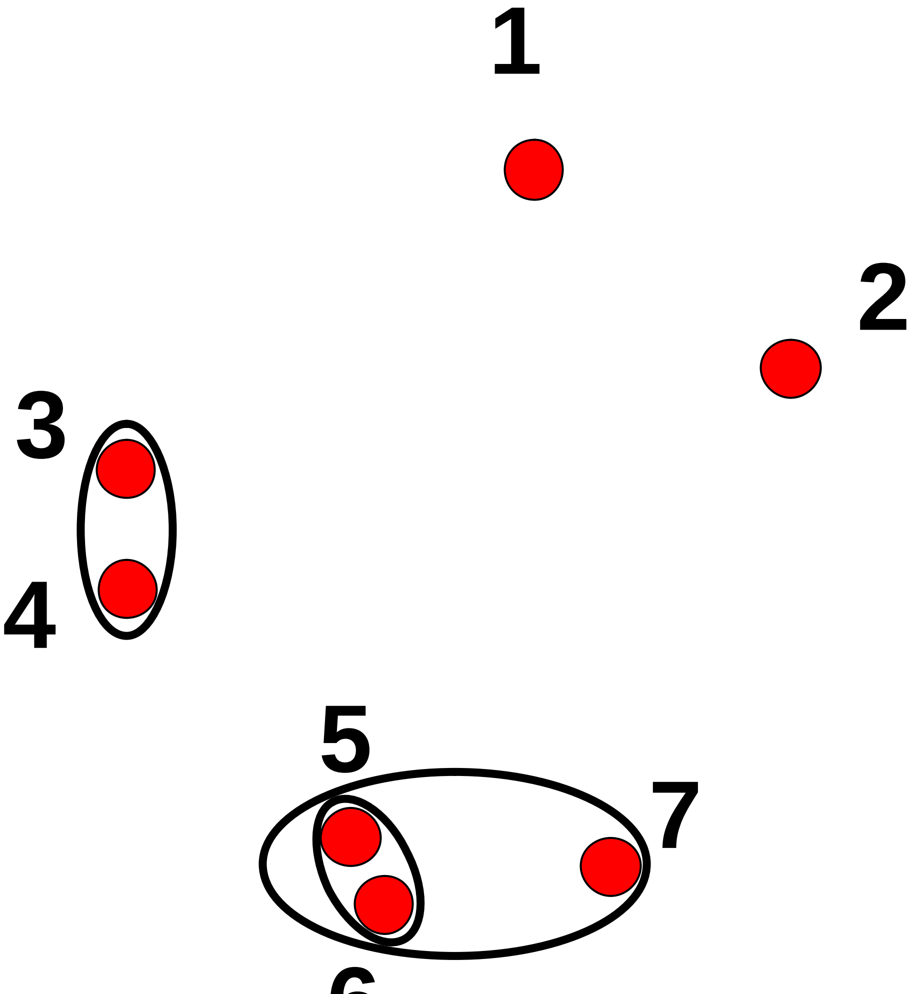
What is clustering?



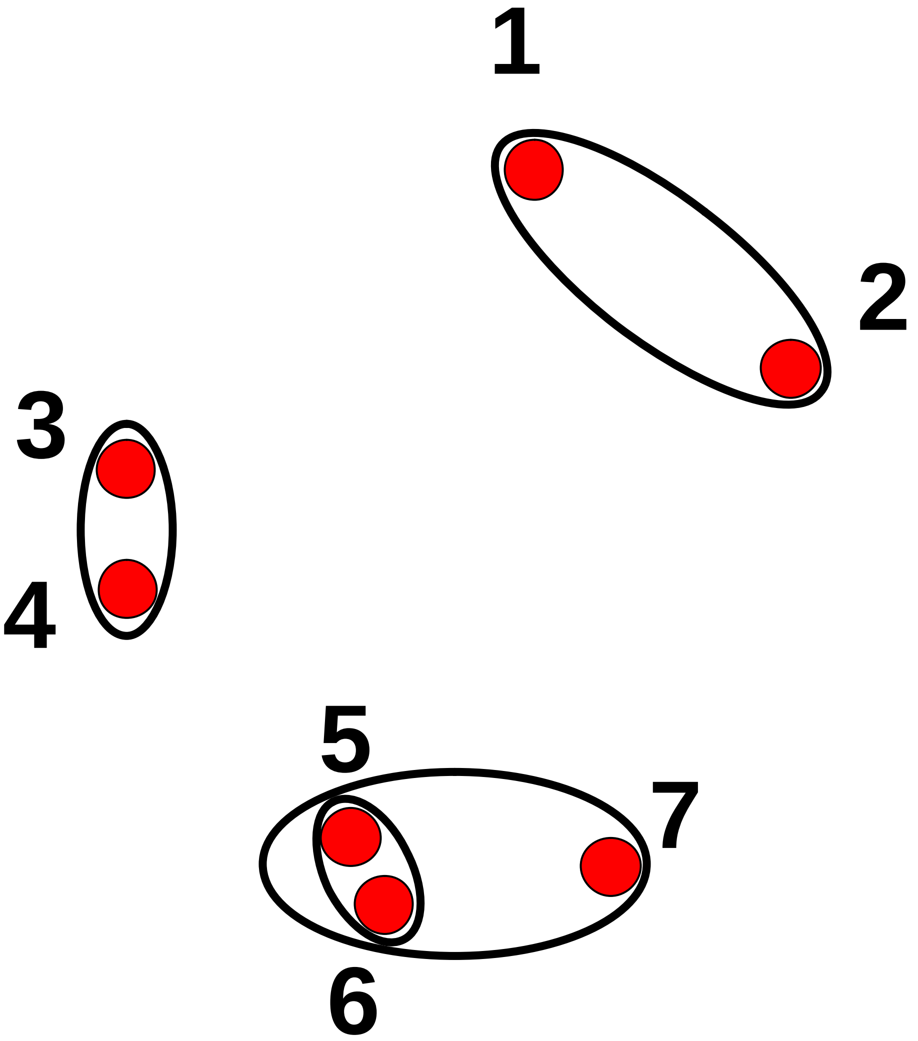
What is clustering?



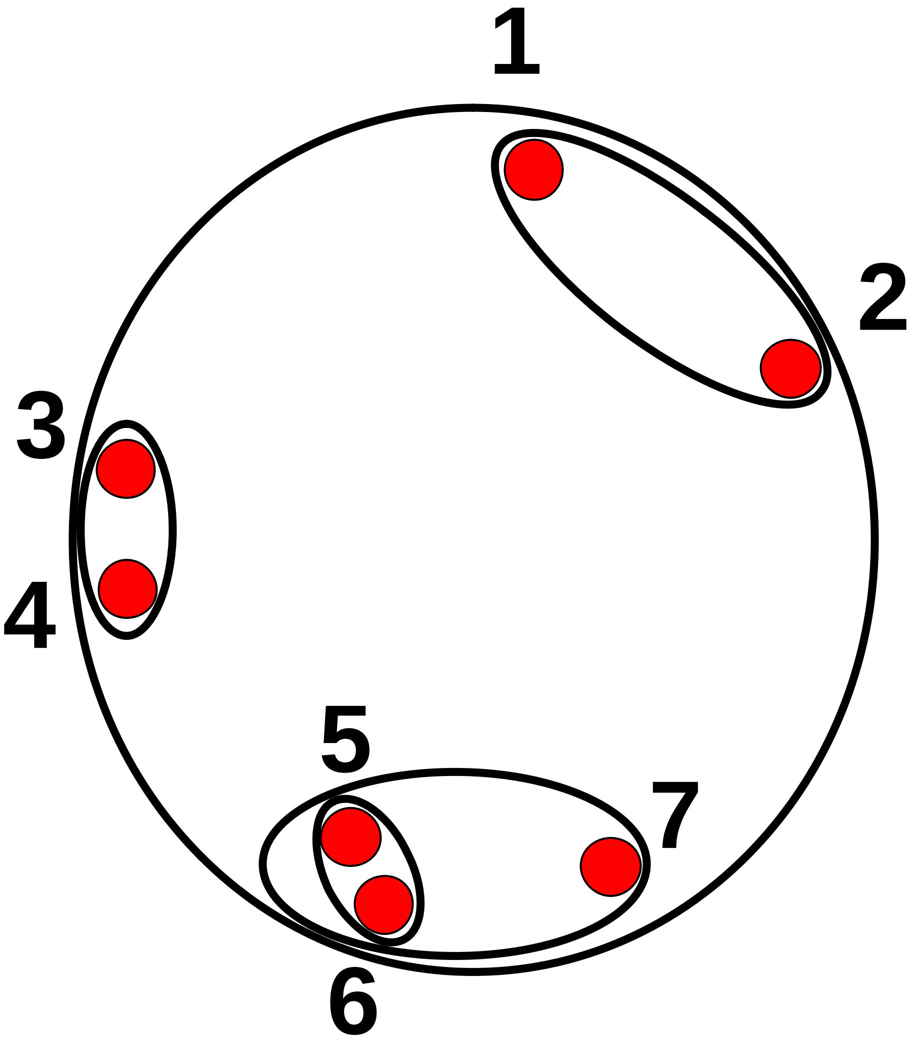
What is clustering?



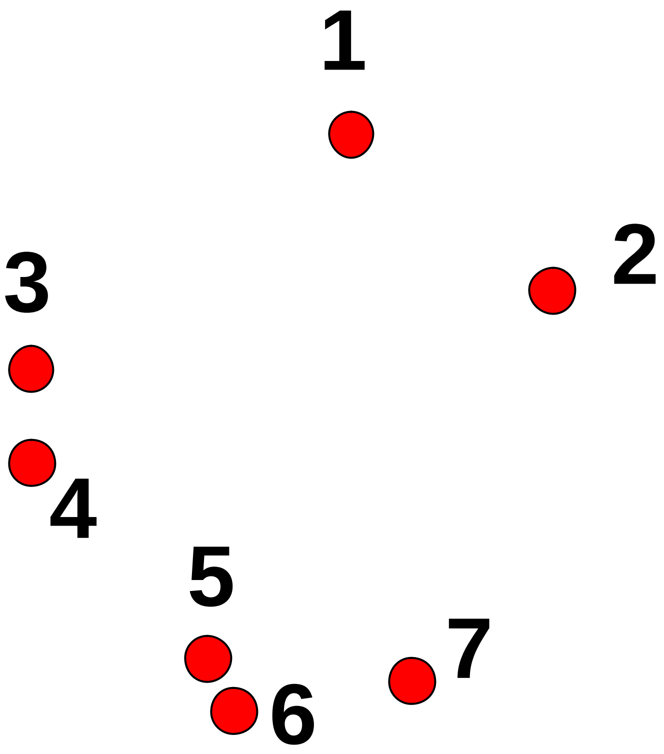
What is clustering?



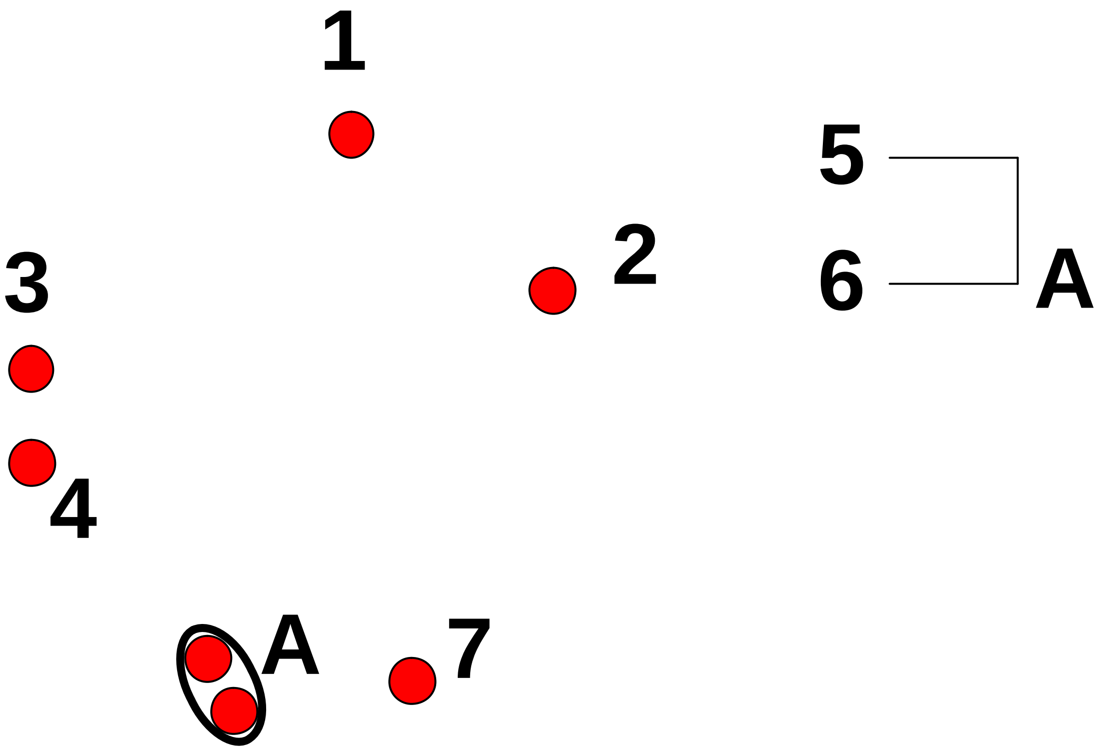
What is clustering?



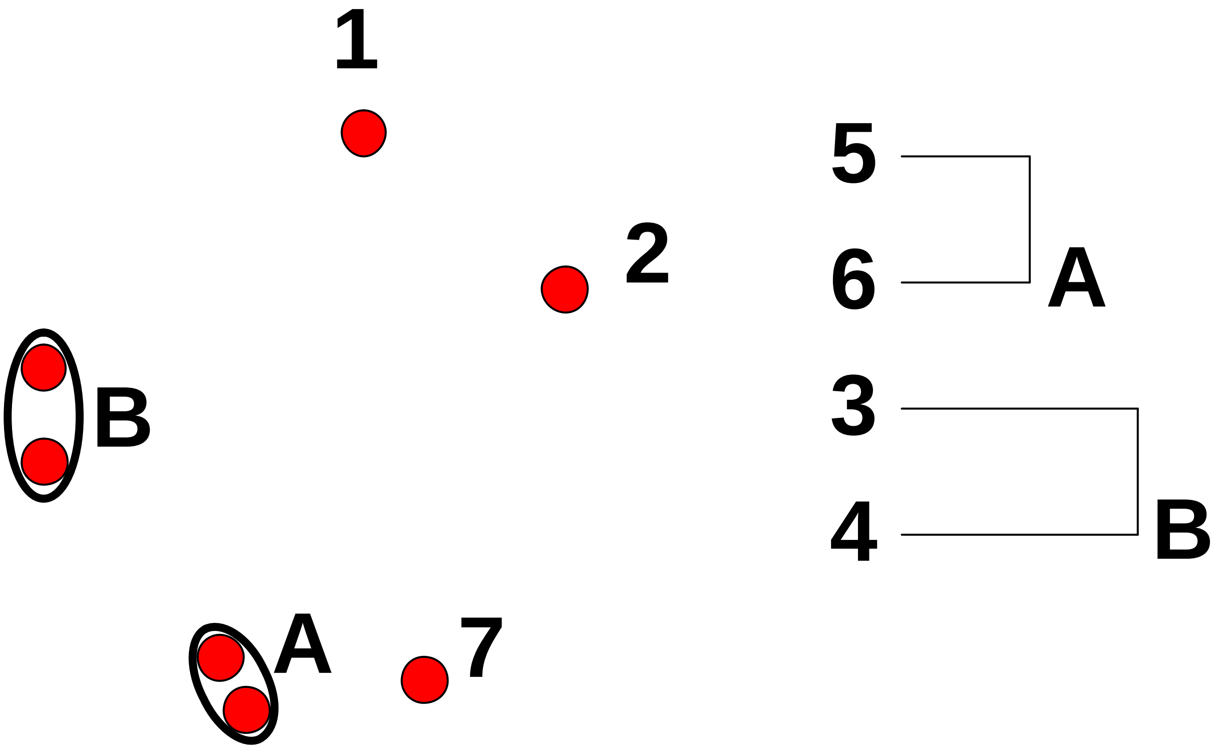
Hierarchical clustering (1)



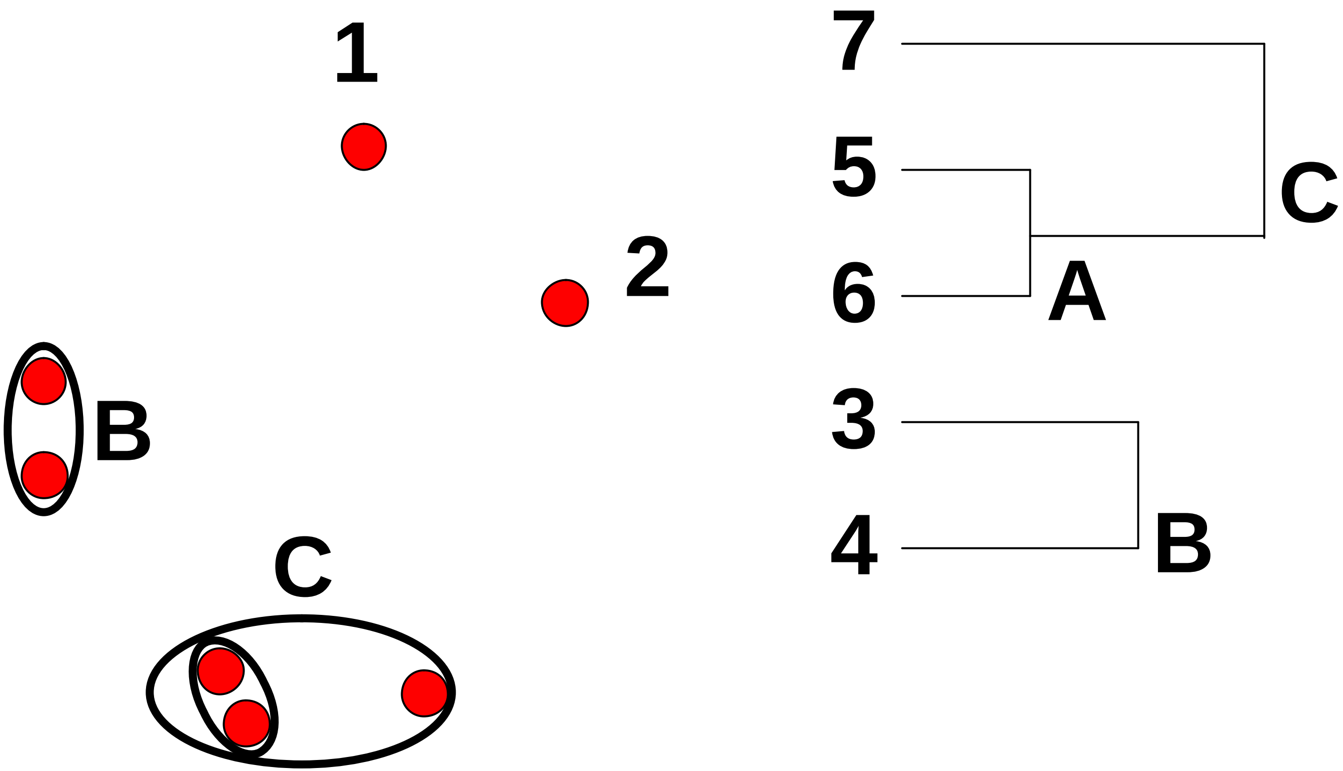
Hierarchical clustering (1)



Hierarchical clustering (1)

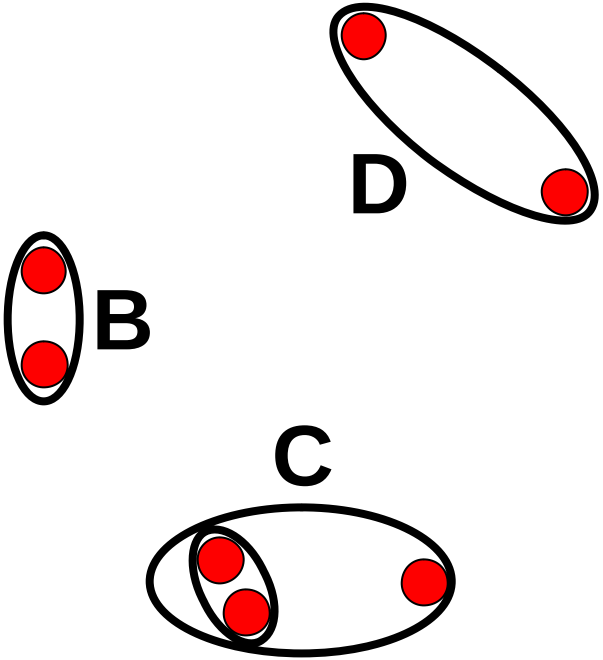


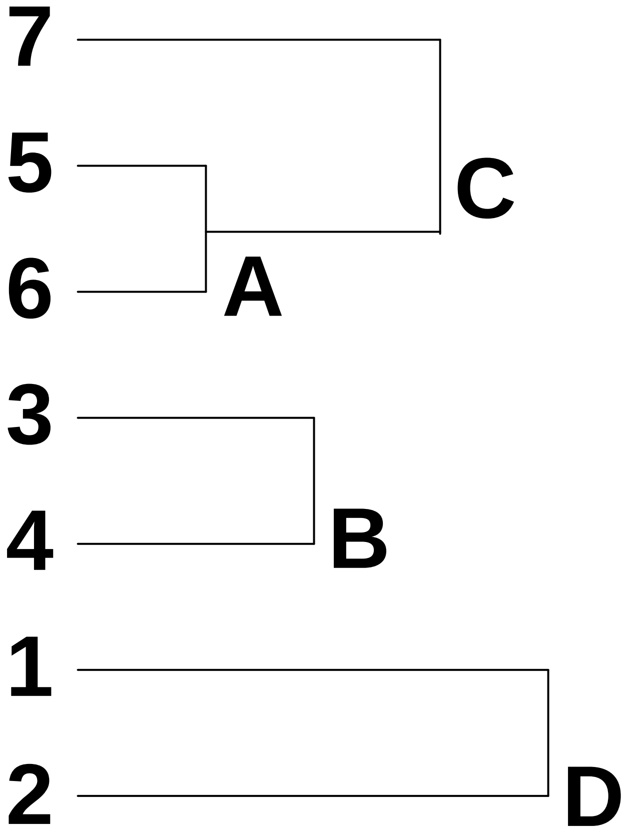
Hierarchical clustering (1)



average linkage

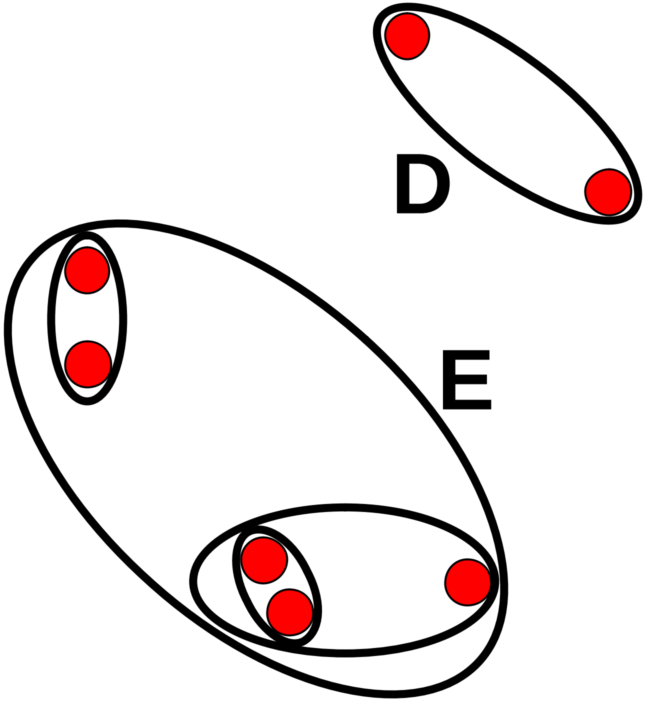
Hierarchical clustering (1)

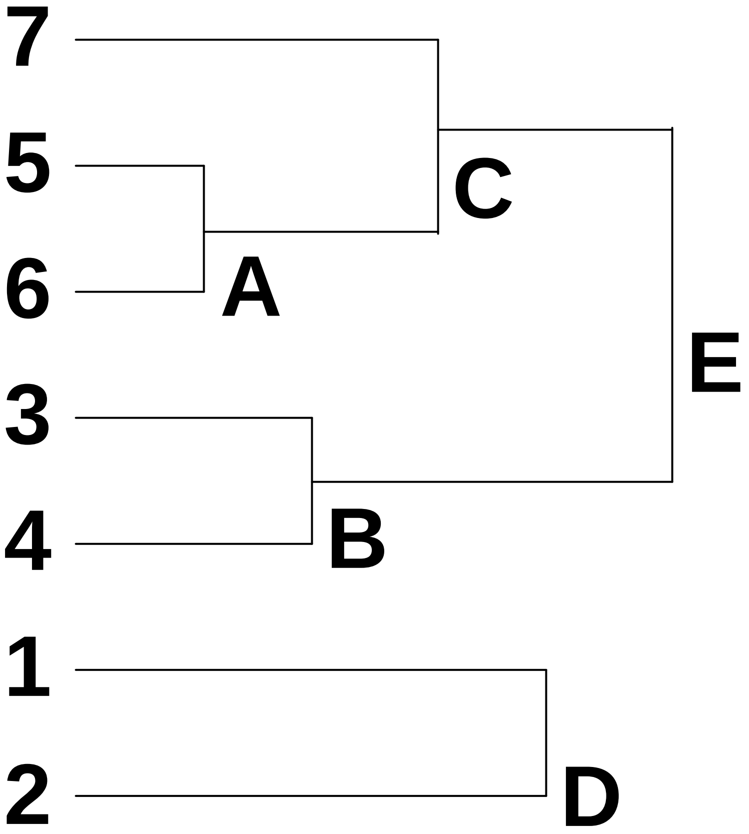




average linkage

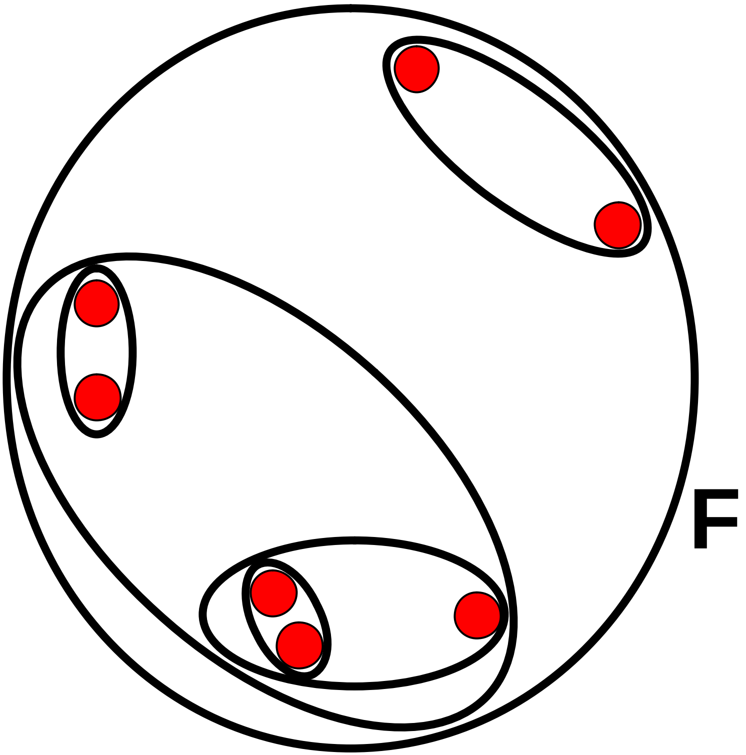
Hierarchical clustering (1)



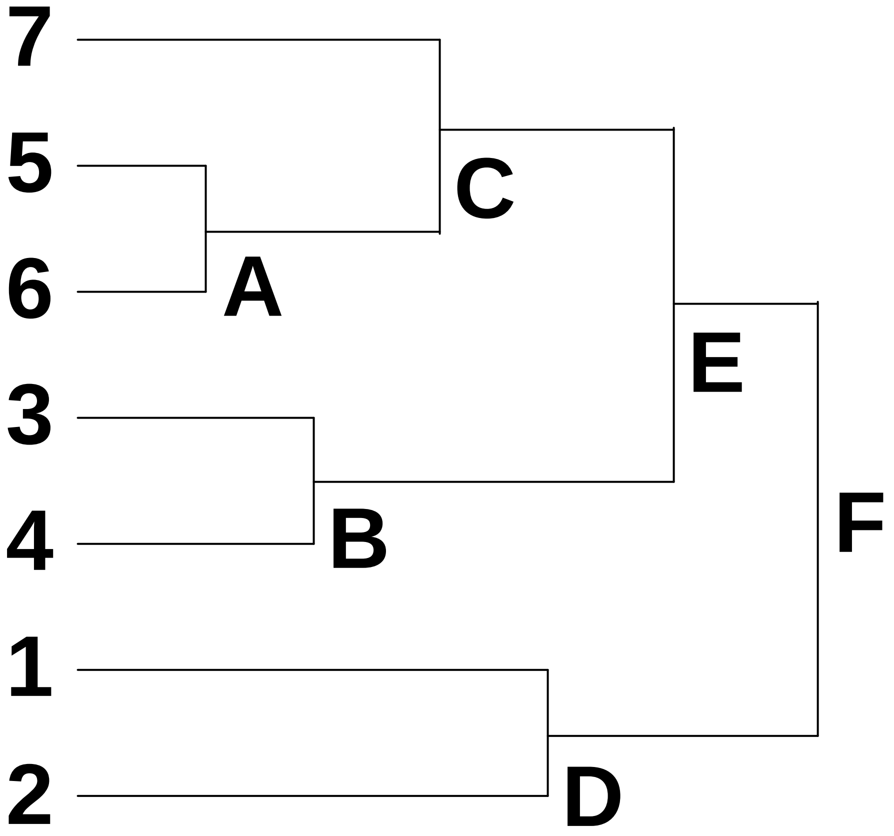


average linkage

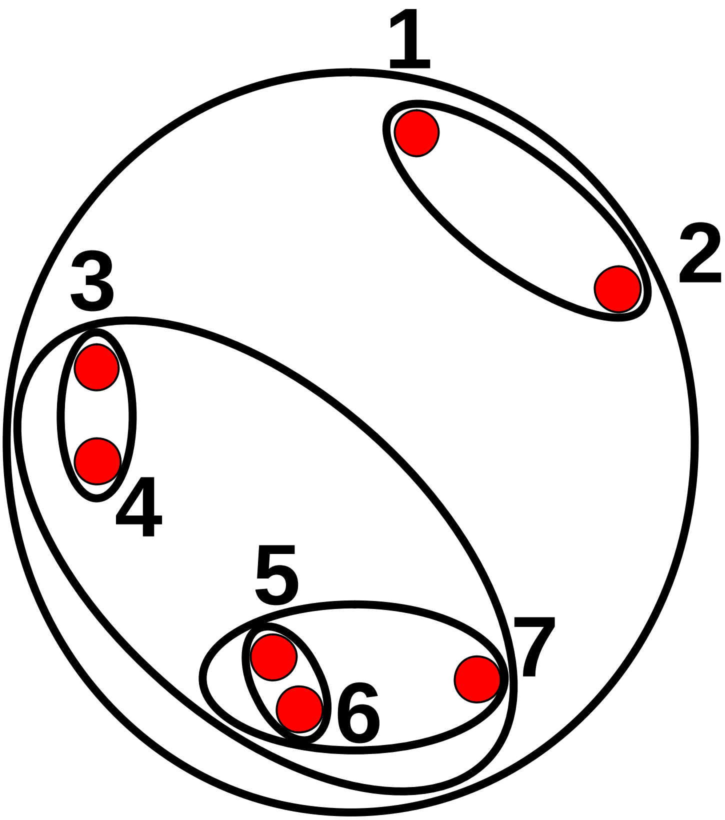
Hierarchical clustering (1)



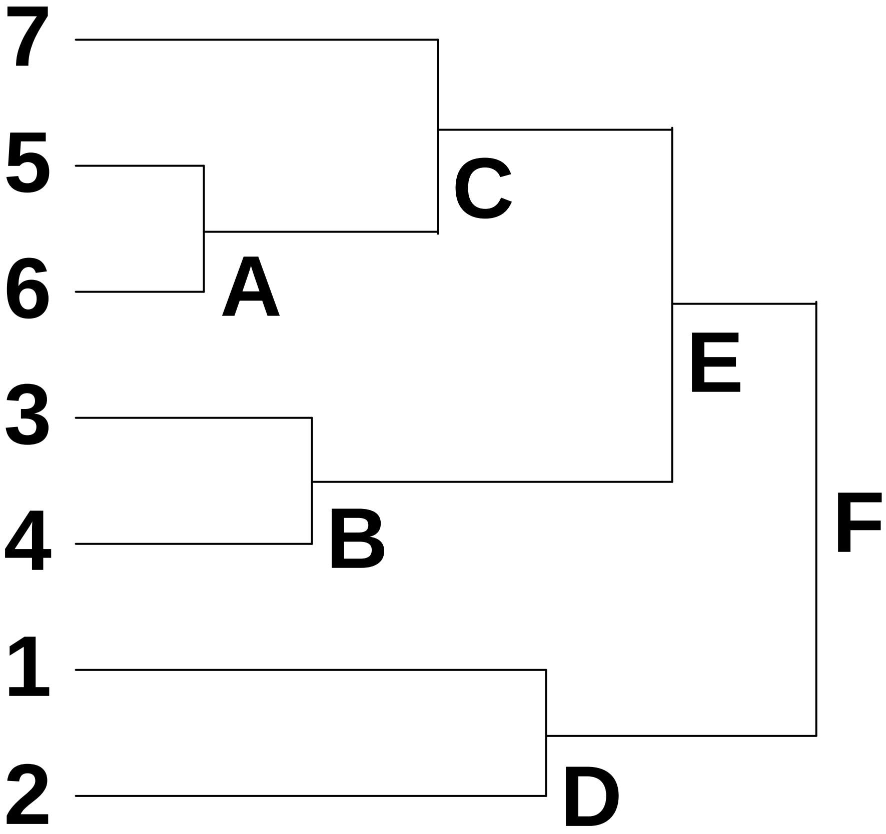
average linkage



Hierarchical clustering (1)

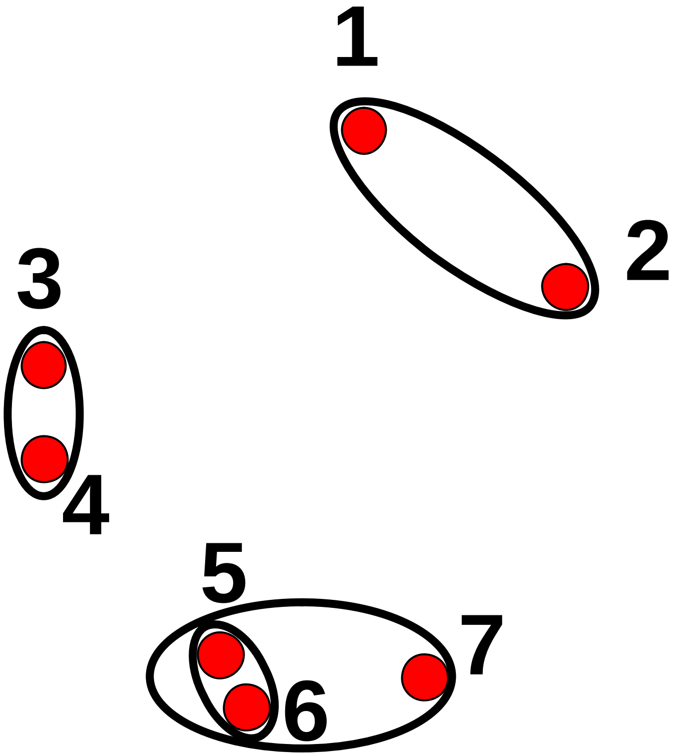


average linkage

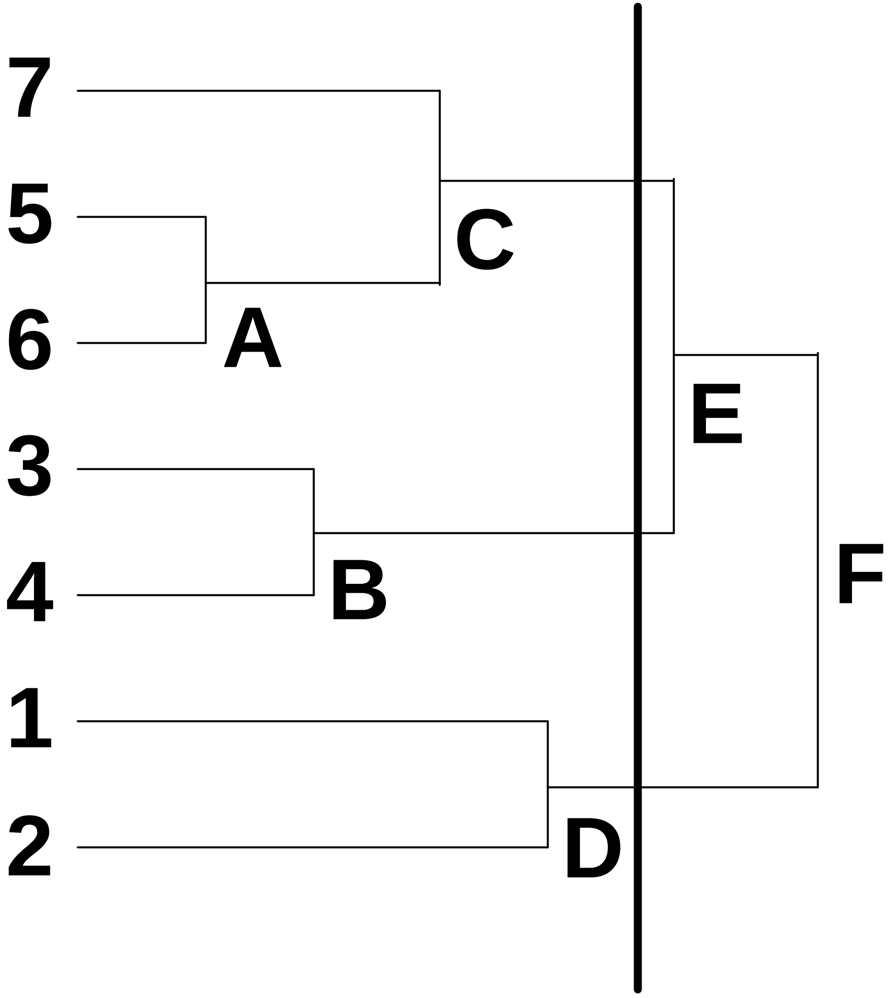


dendrogram

Hierarchical clustering (1)

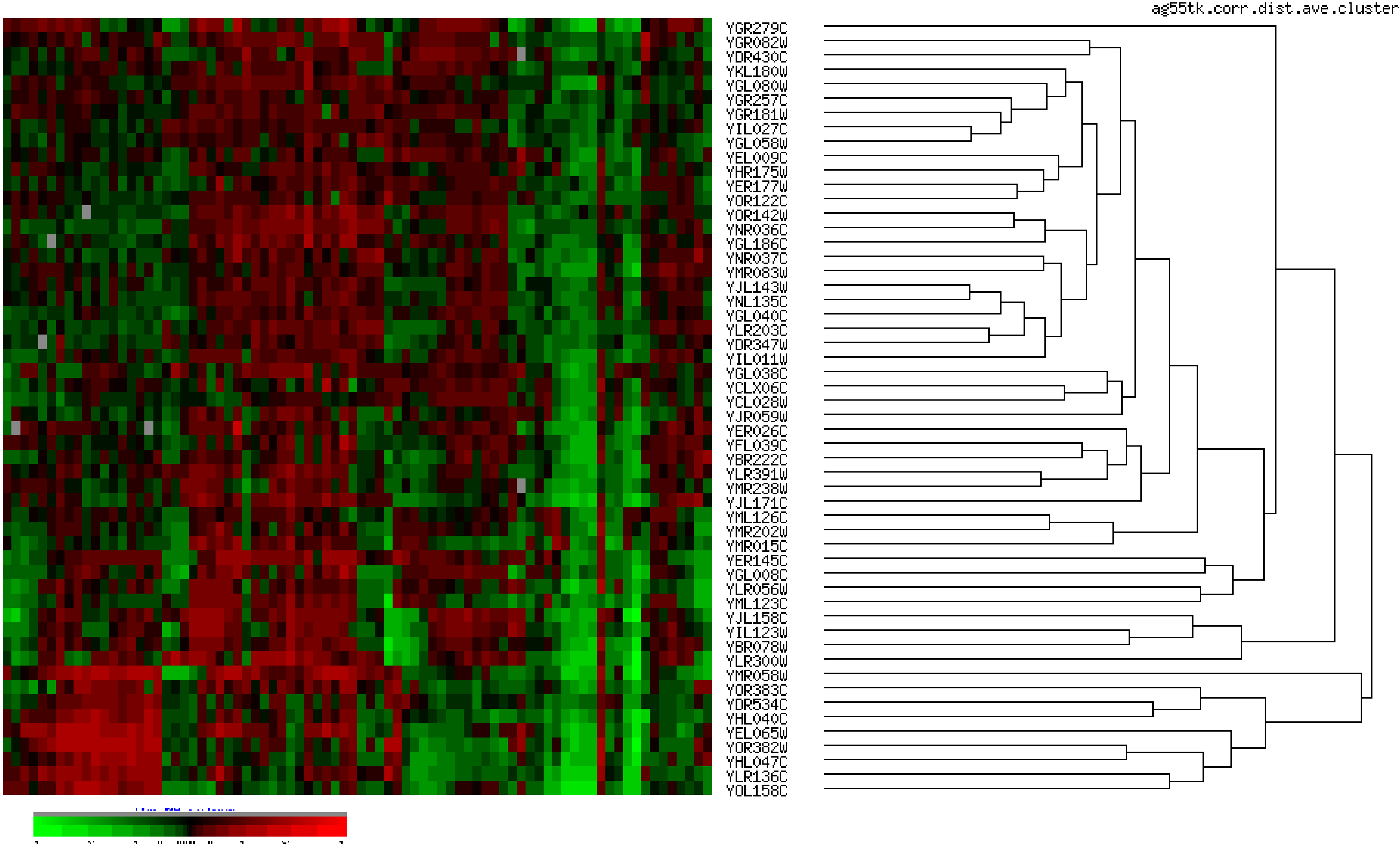


average linkage

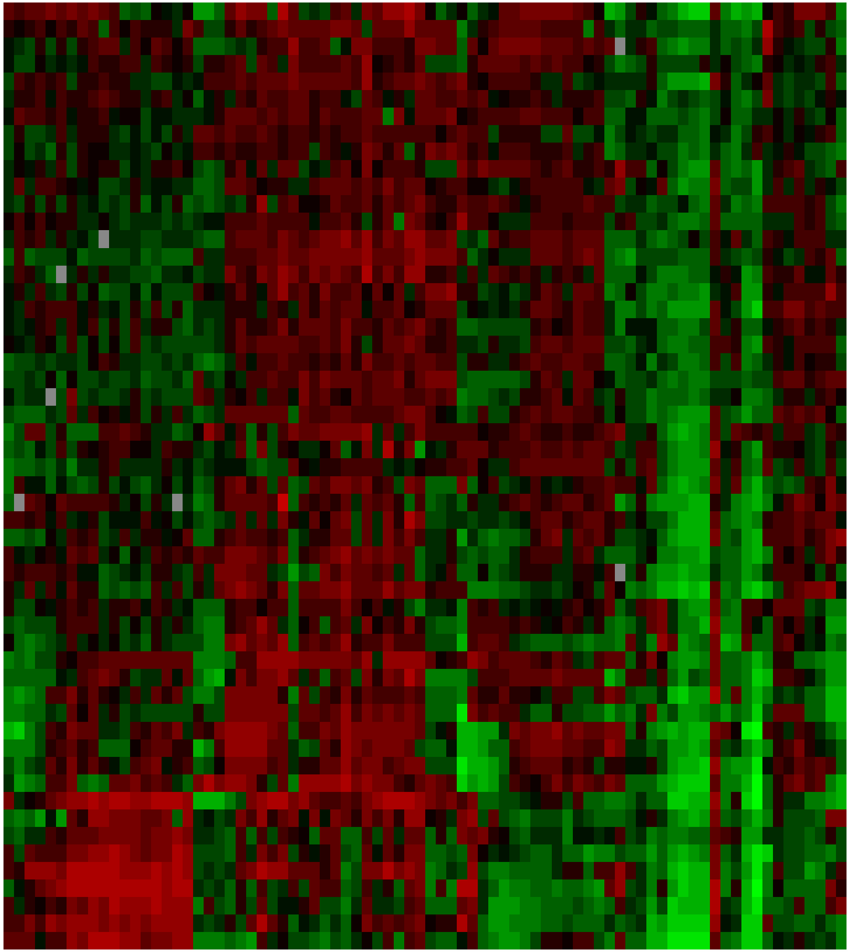


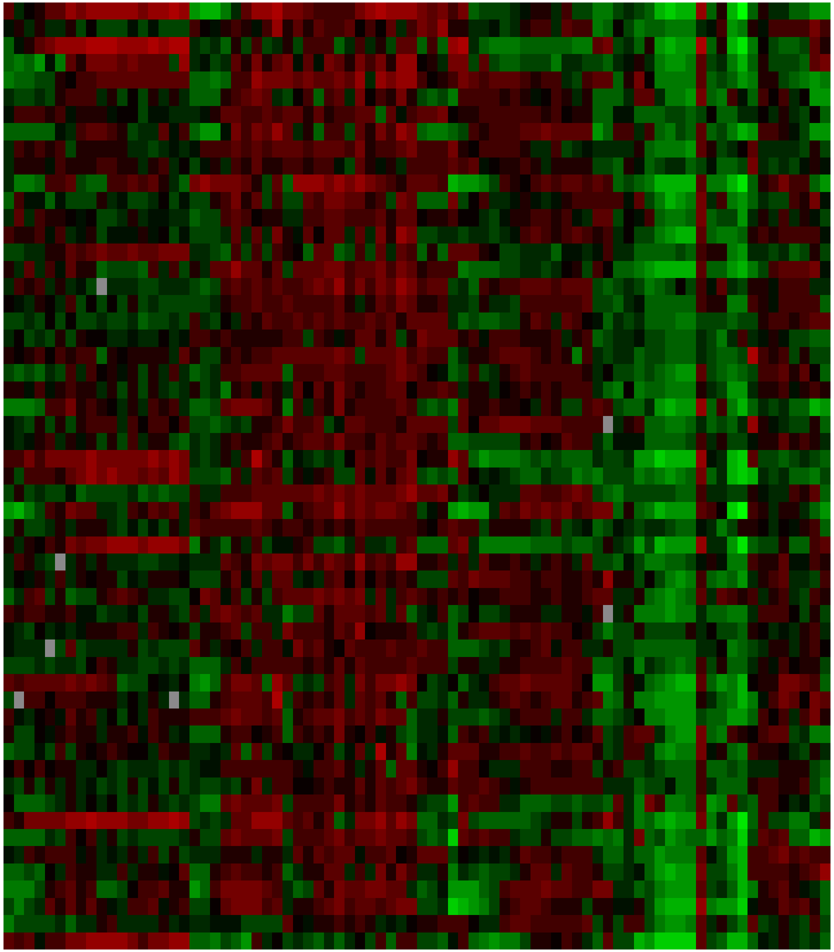
dendrogram

Hierarchical clustering (2)



Hierarchical clustering (3)





Hierarchical clustering (4)

Usually two steps:

- Calculation of all distances: $O(N^2)$ distances
- Agglomerating procedure: $O(N^2)$

Problem:

- N can be large: N>10000

We cannot afford calculating all the distances!

Hierarchical clustering (4)

Usually two steps:

- Calculation of all distances: $O(N^2)$ distances
- Agglomerating procedure: $O(N^2)$

Problem:

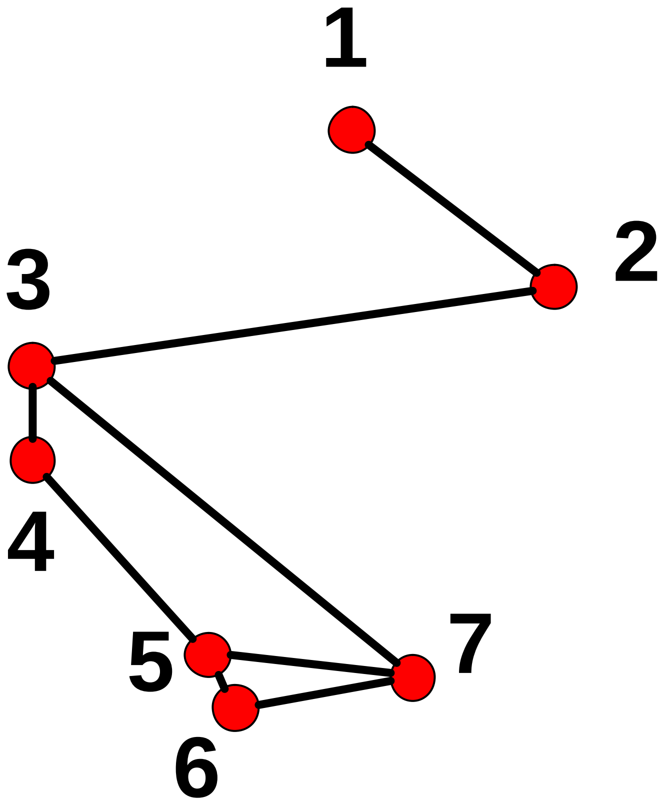
- N can be large: N>10000

We cannot afford calculating all the distances!

What if we calculate only some?

Approximate hierarchical clustering

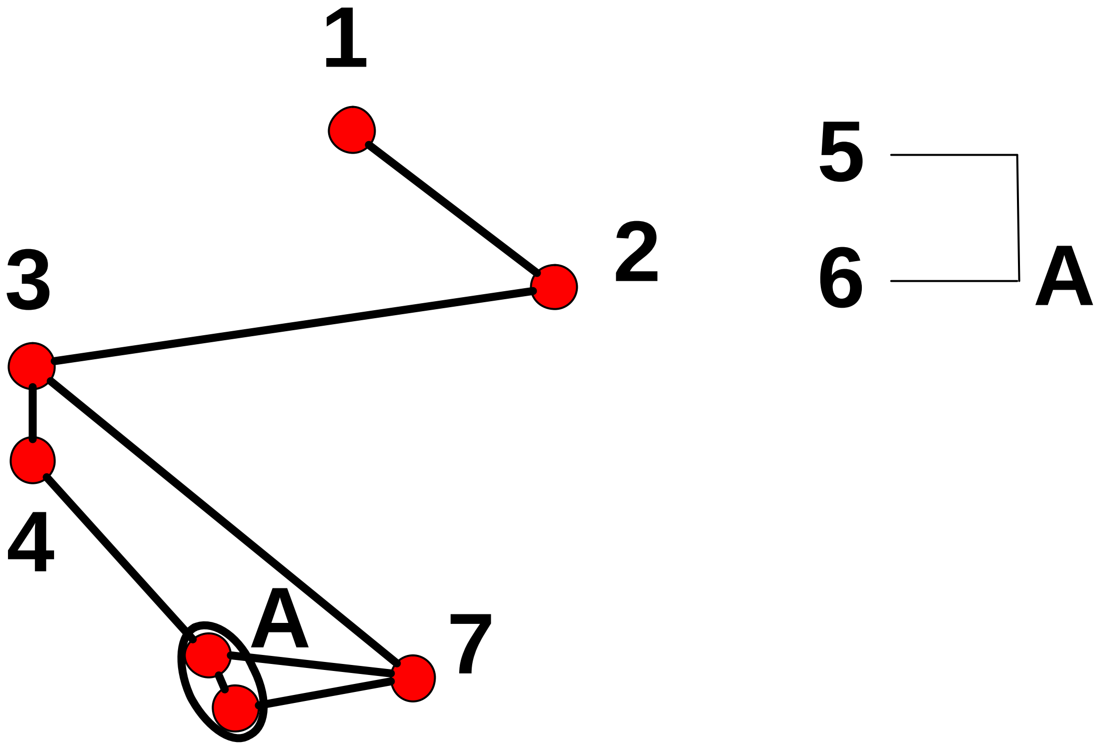
- Calculation of a subset of distances
- Approximate agglomerating procedure



1/3 of distances calculated

Approximate hierarchical clustering

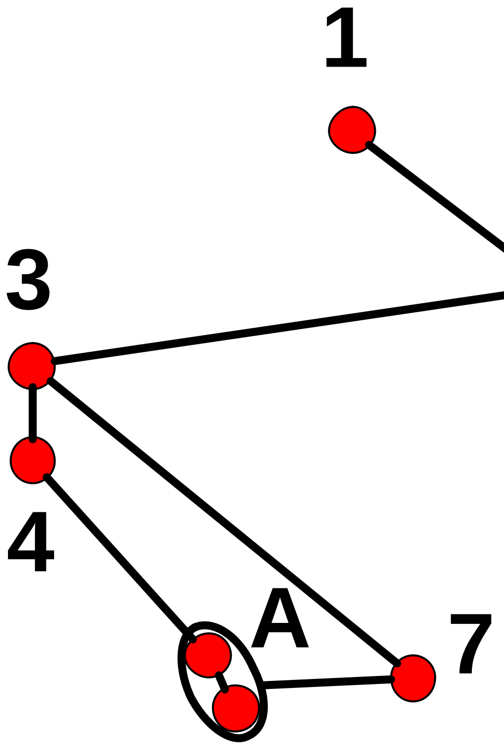
- Calculation of a subset of distances
- Approximate agglomerating procedure

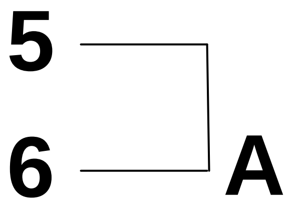


1/3 of distances calculated

Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure

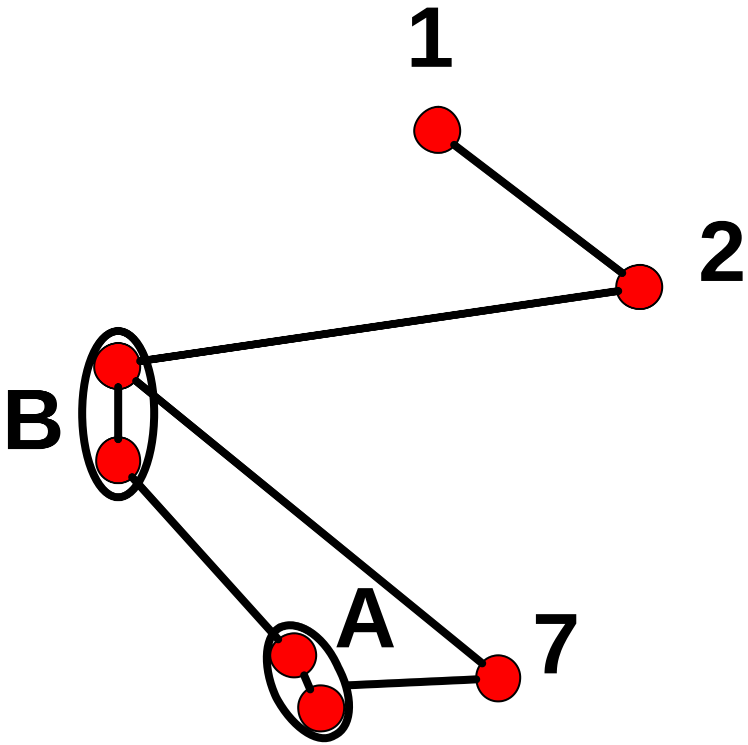




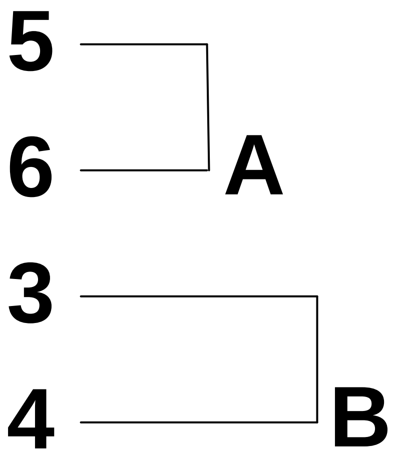
1/3 of distances calculated

Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure

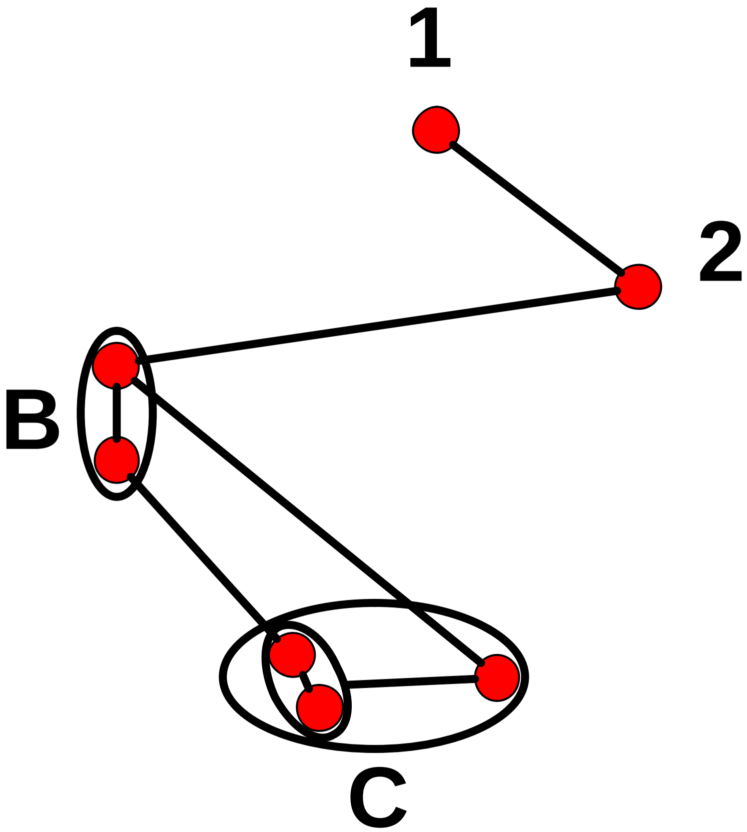


1/3 of distances calculated

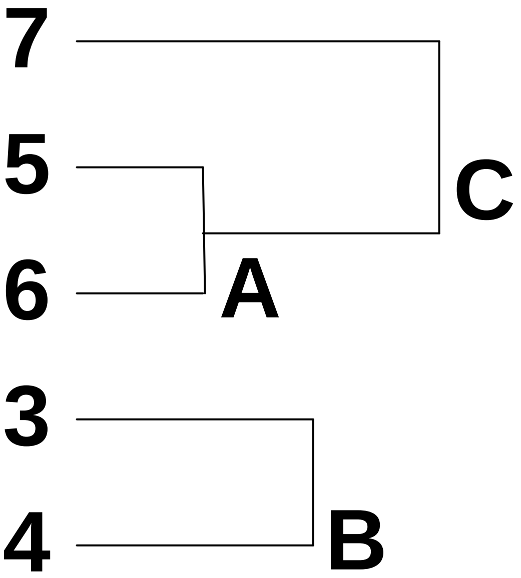


Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure

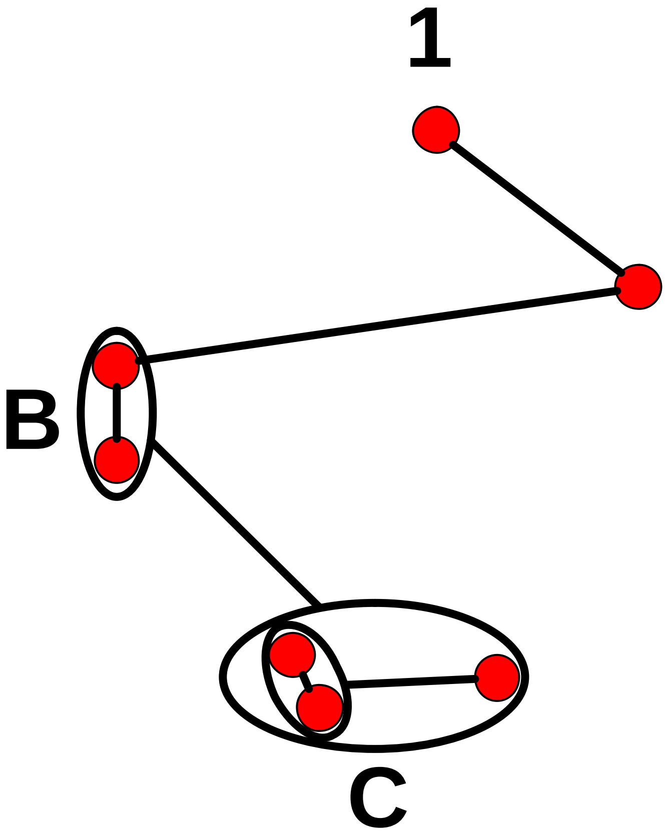


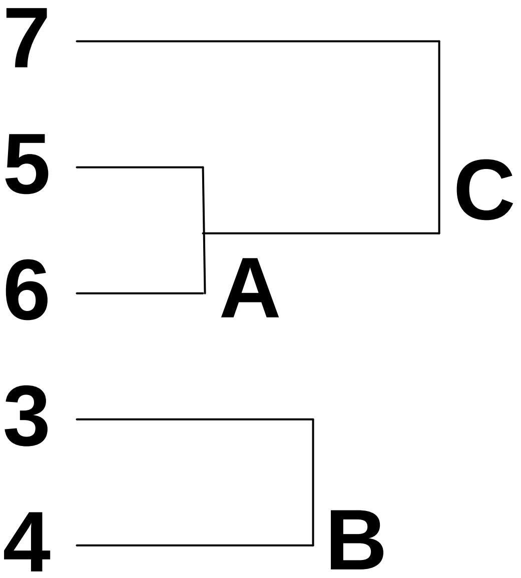
1/3 of distances calculated



Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure

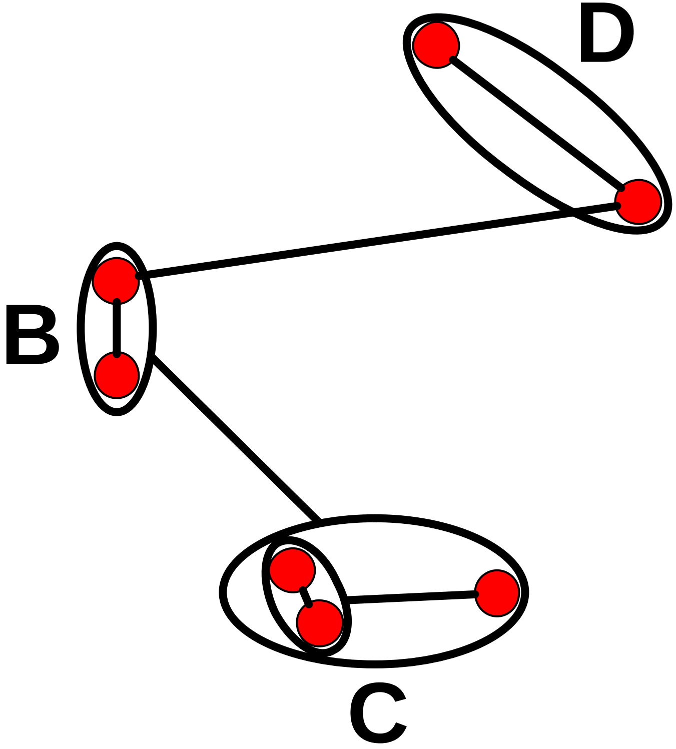




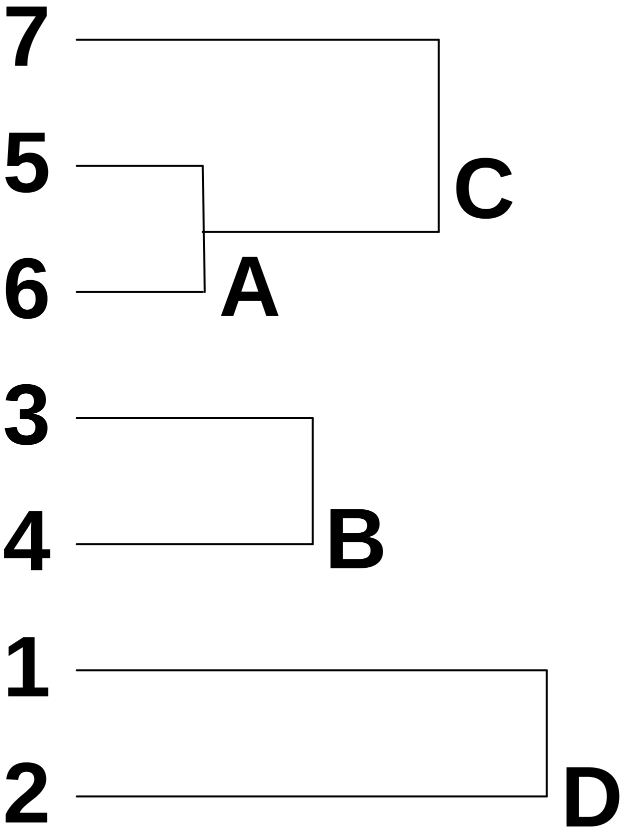
1/3 of distances calculated

Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure

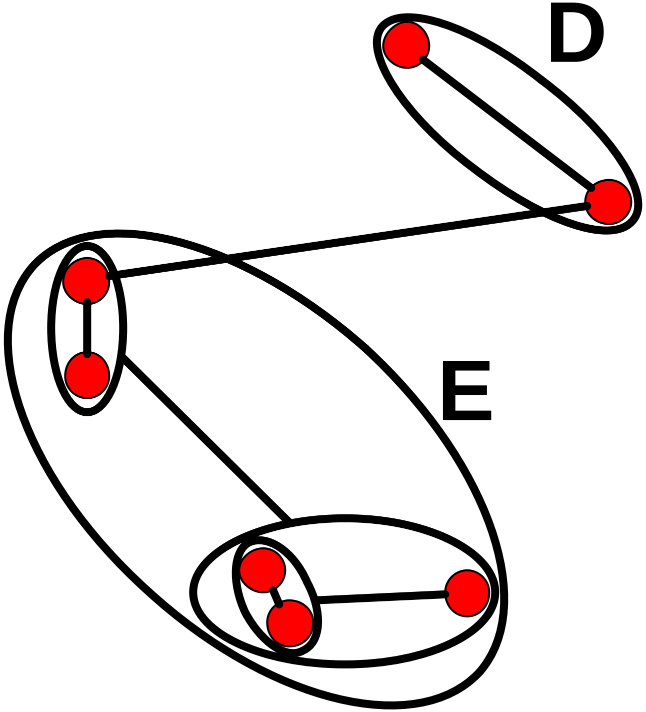


1/3 of distances calculated

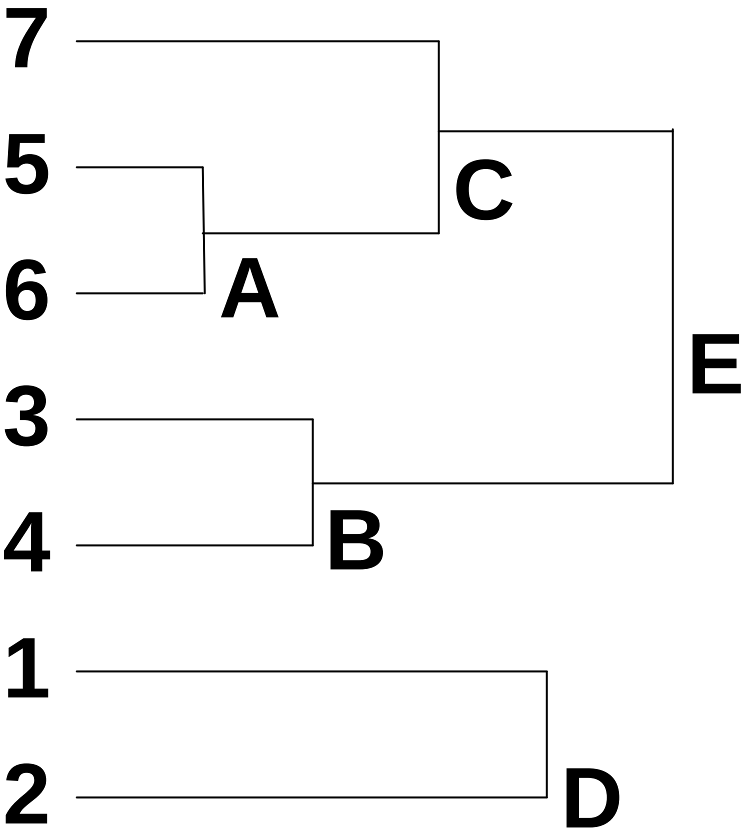


Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure

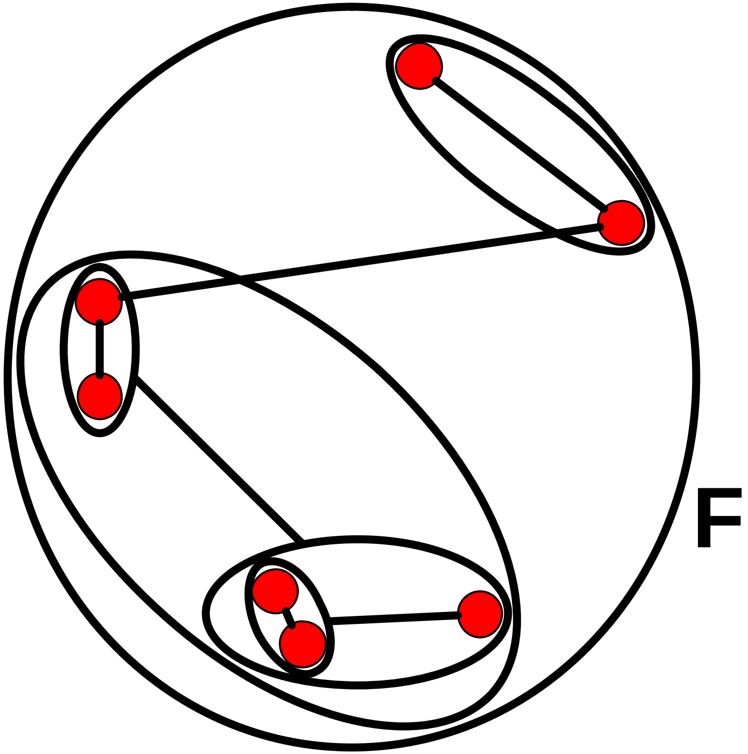


1/3 of distances calculated

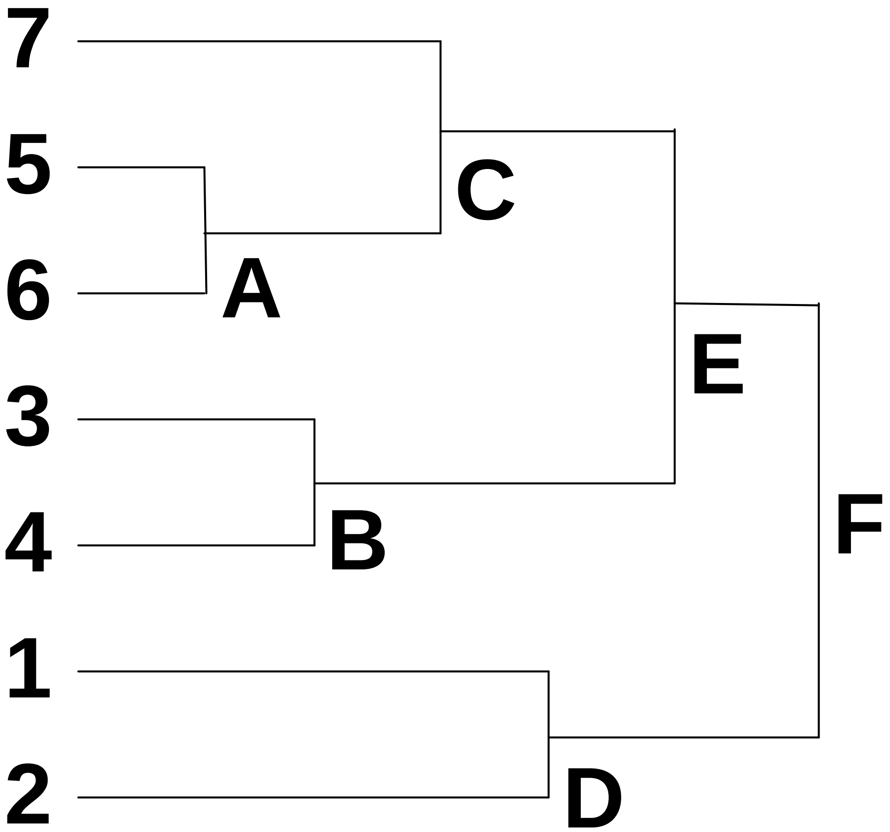


Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure

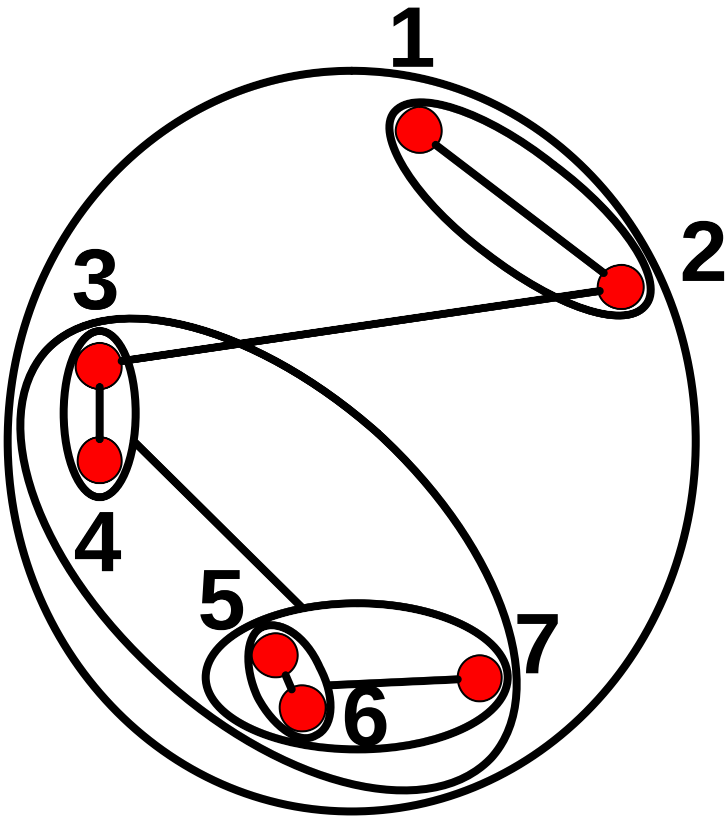


1/3 of distances calculated

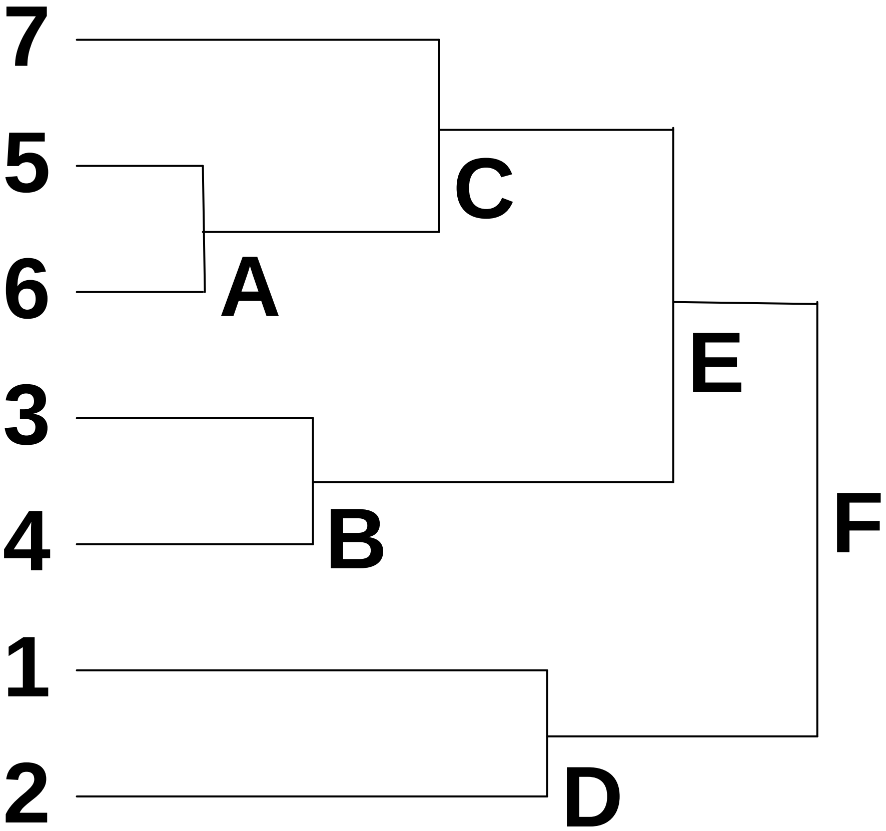


Approximate hierarchical clustering

- Calculation of a subset of distances
- Approximate agglomerating procedure



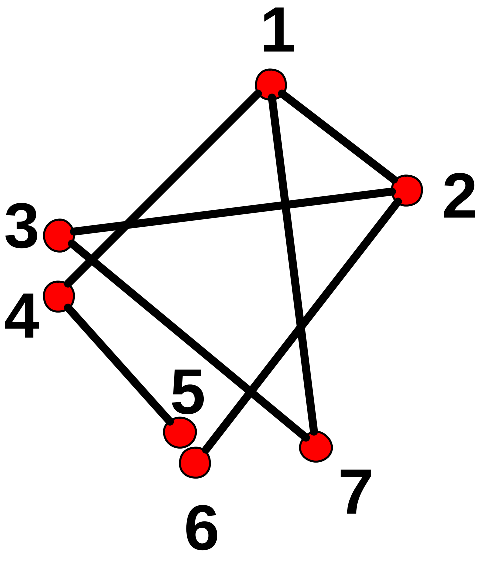
1/3 of distances calculated



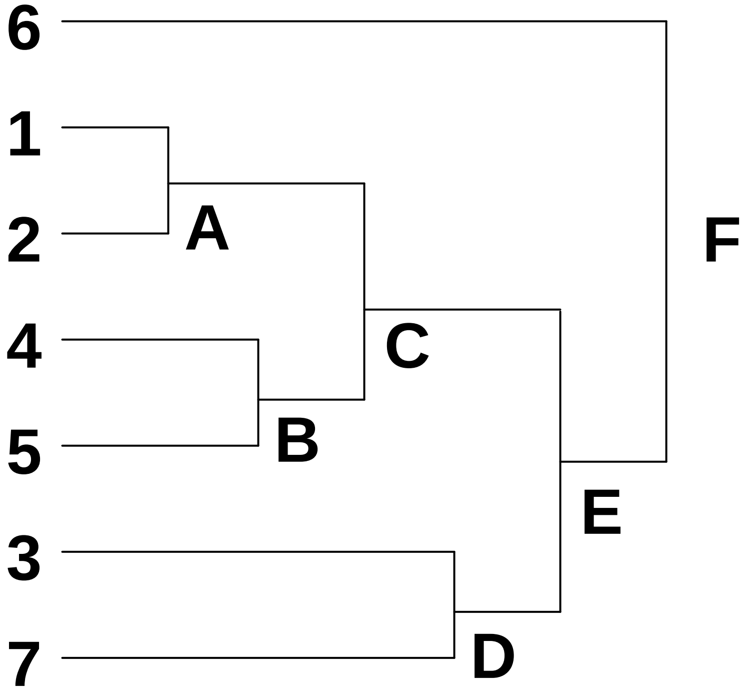
Small distances are important

Results of approximate hierarchical clustering are better when more **distances between similar data items** are used!

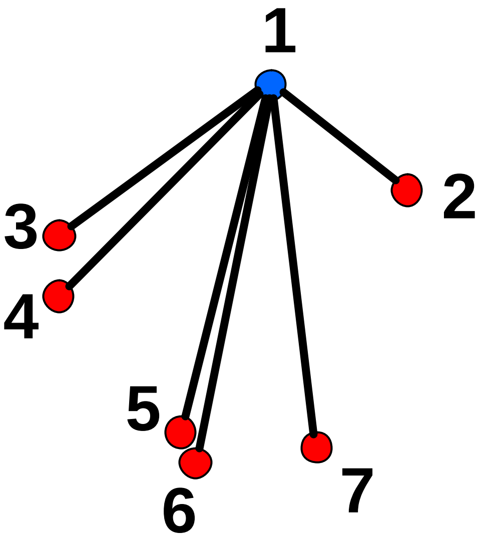
Otherwise:



1/3 of distances calculated



How to find all similar pairs without calculating all the distances?



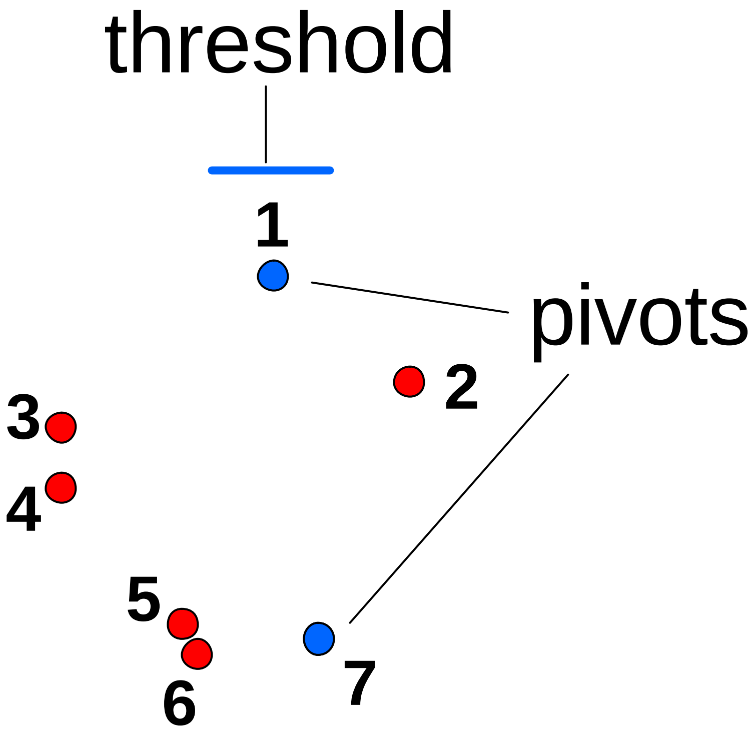
Triangle inequality: d(x,y) <= d(x,z)+d(z,y)

Corollary: d(x,y) >= |d(x,z)-d(y,z)|

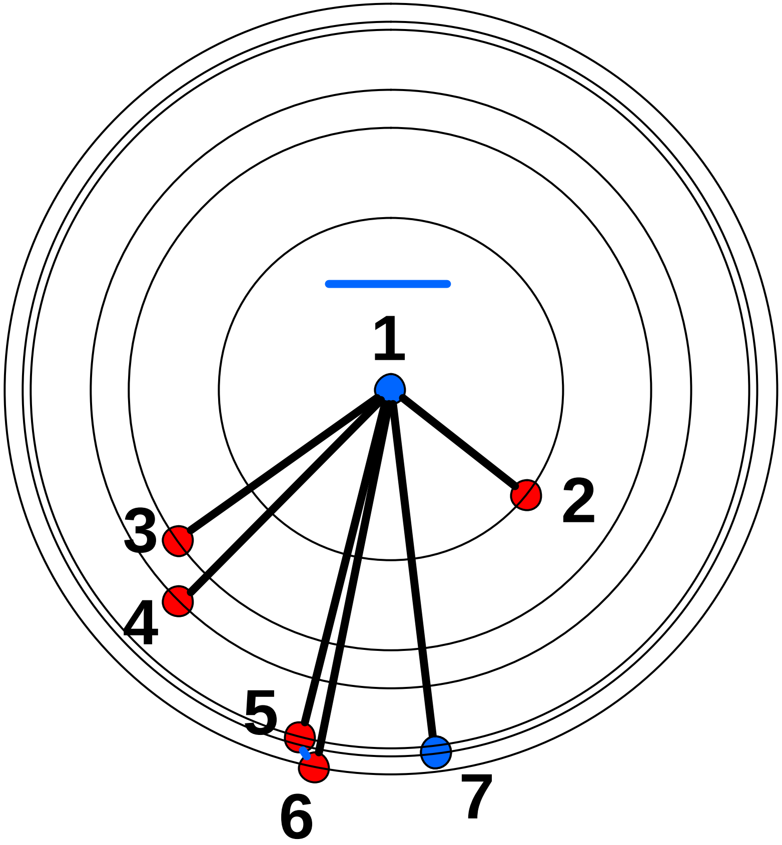
d(**5**,**2**) >= |d(**1**,**5**)-d(**1**,**2**)|

d(**5**,**6**) >= |d(**1**,**5**)-d(**1**,**6**)|

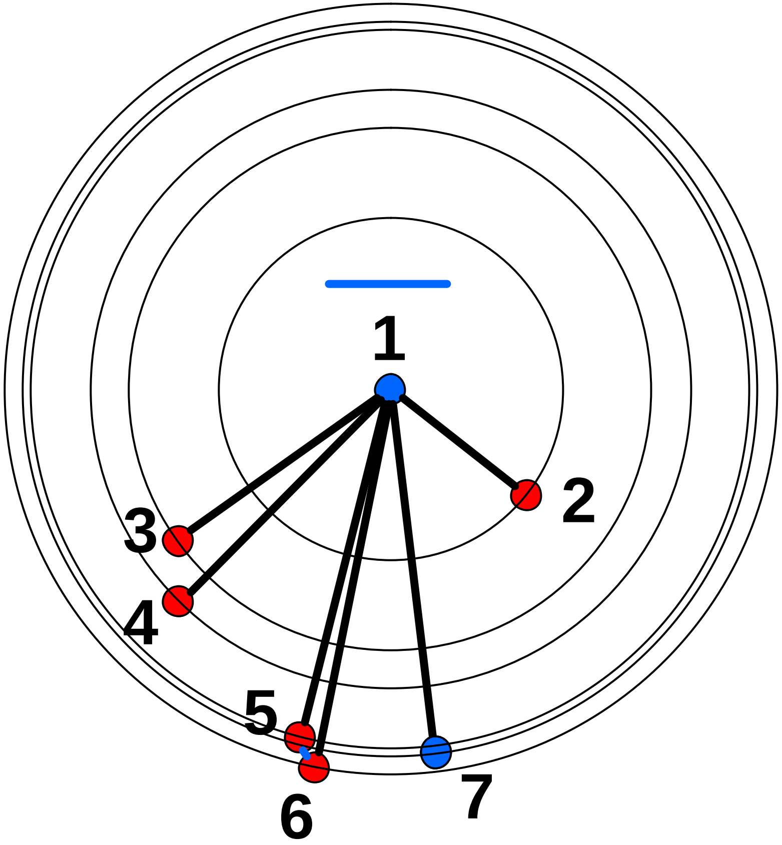
Closest pairs algorithm



Closest pairs algorithm



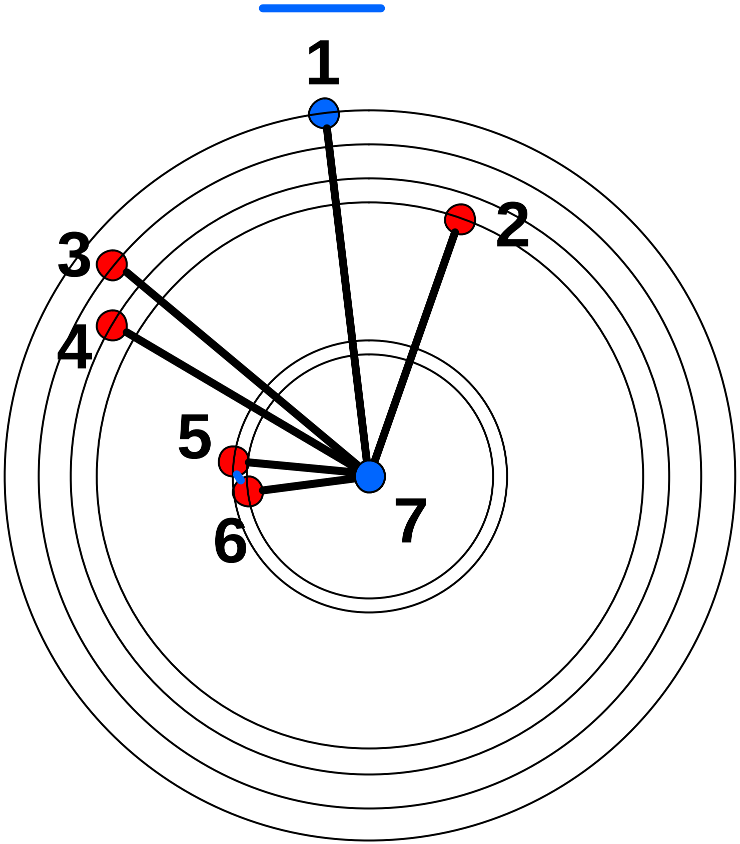
Closest pairs algorithm



Candidates from 1:

- 2-3
- 3-4
- 3-5
- 4-5
- 4-6
- 4-7
- 5-6
- 5-7
- 6-7

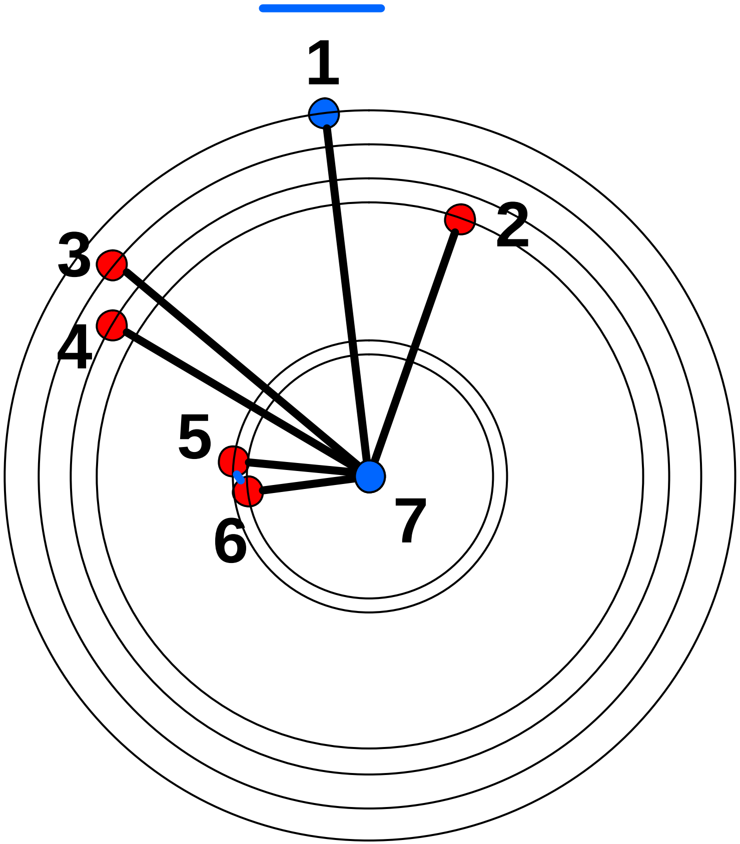
Closest pairs algorithm



Candidates from 1:

- 2-3
- 3-4
- 3-5
- 4-5
- 4-6
- 4-7
- 5-6
- 5-7
- 6-7

Closest pairs algorithm



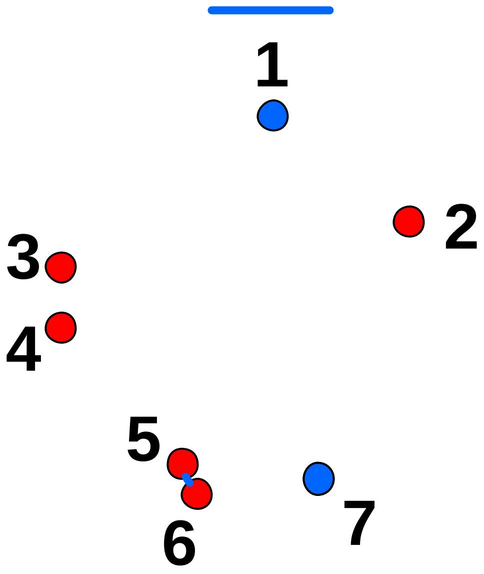
Candidates from 1:

- 2-3
- 3-4
- 3-5
- 4-5
- 4-6
- 4-7
- 5-6
- 5-7
- 6-7

Candidates from 7:

- 1-2
- 1-3
- 1-4
- 2-3
- 2-4
- 3-4
- 5-6
- 5-7
- 6-7

Closest pairs algorithm



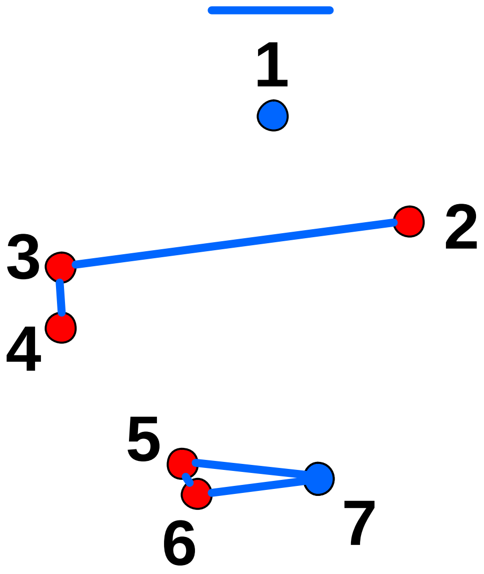
Candidates from 1:

- 2-3
- 3-4
- 3-5
- 4-5
- 4-6
- 4-7
- 5-6
- 5-7
- 6-7

Candidates from 7:

- 1-2
- 1-3
- 1-4
- 2-3
- 2-4
- 3-4
- 5-6
- 5-7
- 6-7

Closest pairs algorithm



Candidates from 1:

- 2-3
- 3-4
- 3-5
- 4-5
- 4-6
- 4-7
- 5-6
- 5-7
- 6-7

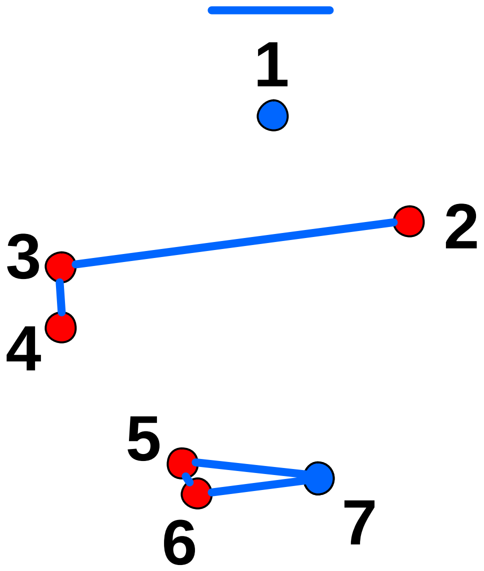
Candidates from 7:

- 1-2
- 1-3
- 1-4
- 2-3
- 2-4
- 3-4
- 5-6
- 5-7
- 6-7

Final candidates:

- 2-3
- 3-4
- 5-6
- 5-7
- 6-7

Closest pairs algorithm



Candidates from 1:

- 2-3
- 3-4
- 3-5
- 4-5
- 4-6
- 4-7
- 5-6
- 5-7
- 6-7

Candidates from 7:

- 1-2
- 1-3
- 1-4
- 2-3
- 2-4
- 3-4
- 5-6
- 5-7
- 6-7

Final candidates:

- 2-3
- 3-4 +
- 5-6 +
- 5-7
- 6-7

Example continued

Dataset:

- N = **7** datapoints;
- N*(N-1)/2 = **21** pairs of datapoints.

Closest pairs algorithm

- calculated **15** distances, it means ~**70%**;
- found the **4** closest pairs (**2** of them closer than threshold).

15 randomly chosen distances contain on

the average **3** of these 4 similar pairs.

Large example

Dataset:

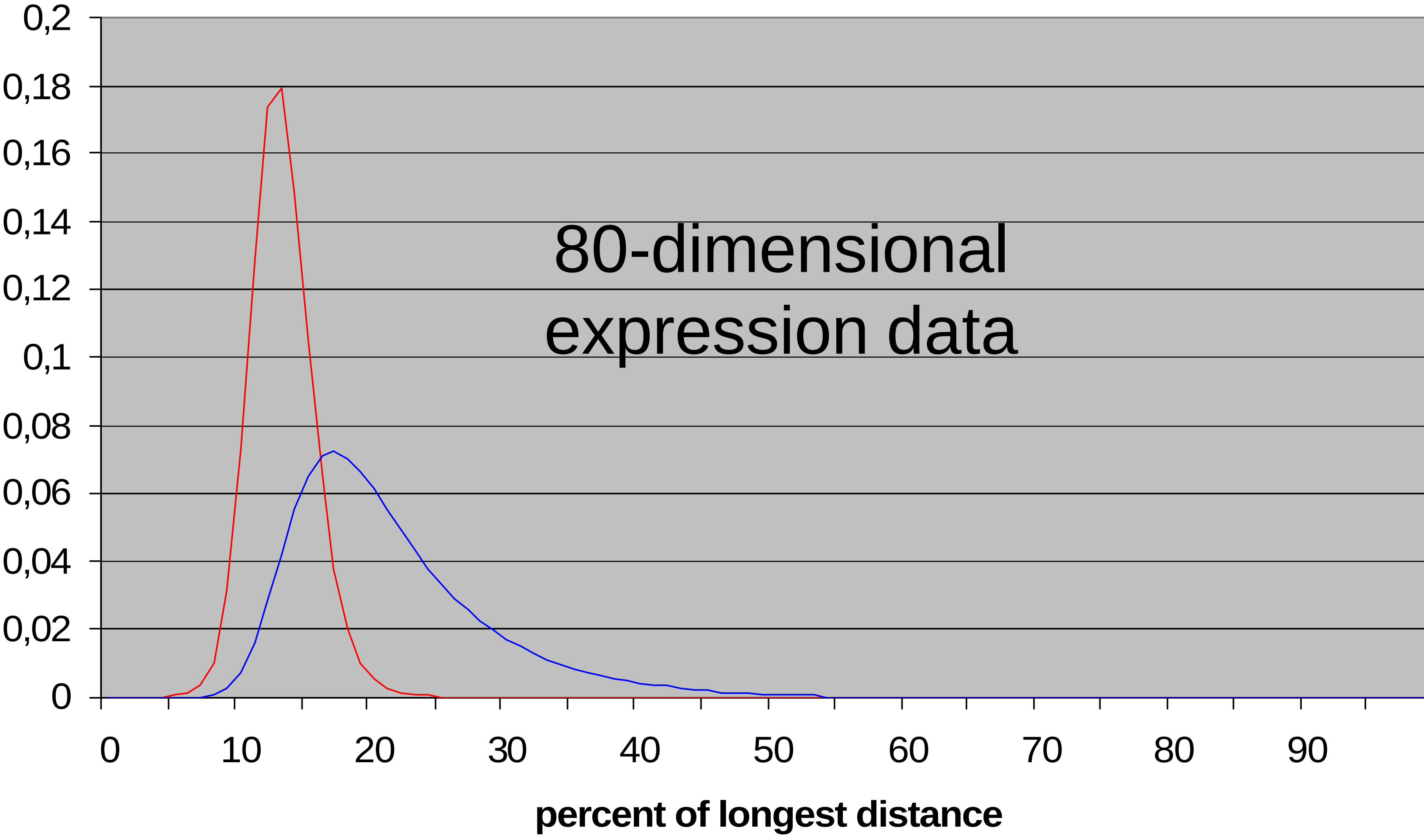
- N = **6000** datapoints;
- N*(N-1)/2 ~ **18 million** pairs of datapoints.

Closest pairs algorithm

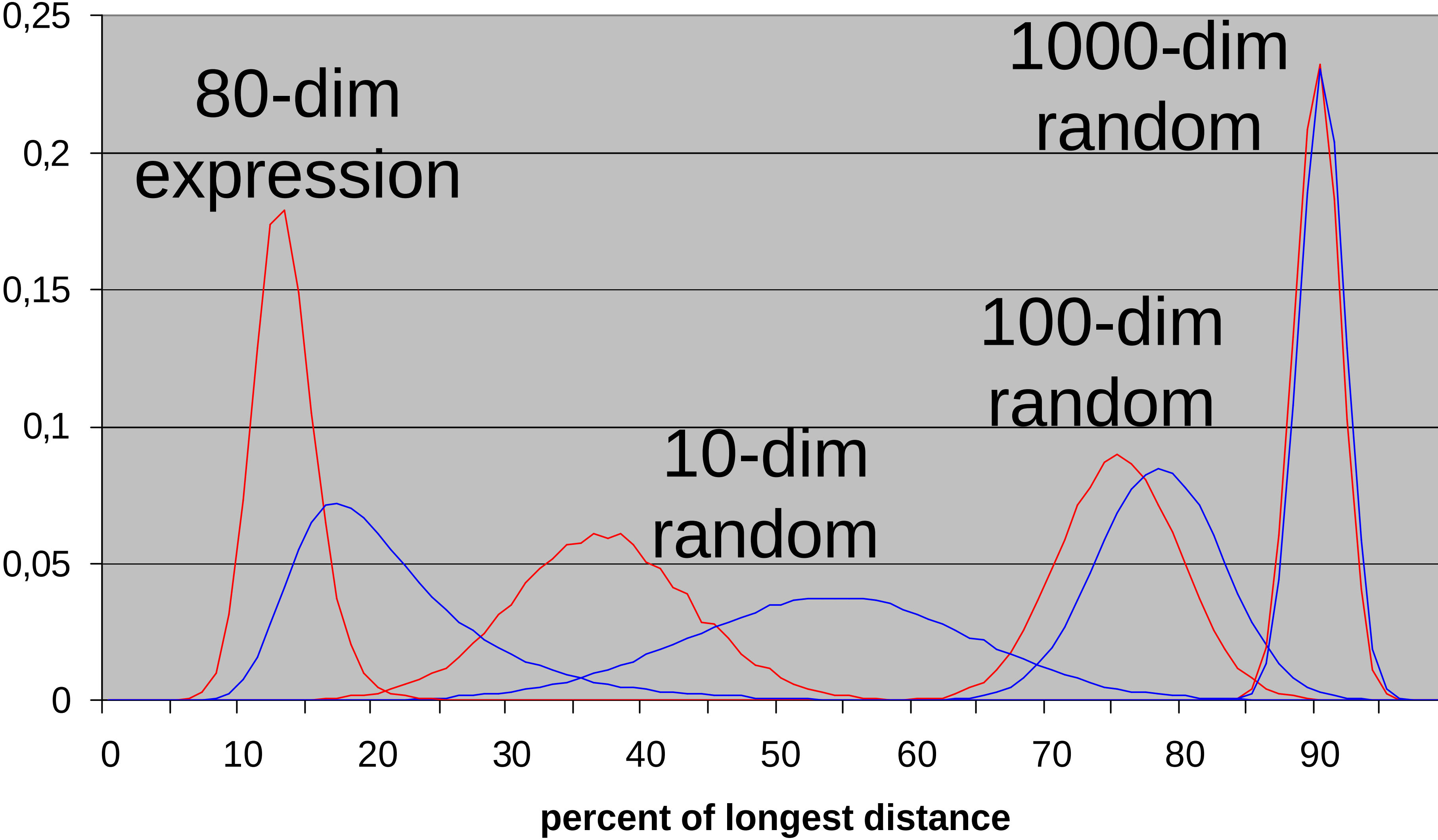
- calculated **1.4 million** distances, i.e. ~**7%**;
- found the **10000** closest pairs (**1000** of them closer than threshold).

1.4 million randomly chosen distances contain on the average only **700** of these 10000 pairs.

Probability distributions of all distances vs distances from the closest pairs algorithm



Probability distributions of all distances vs distances from the closest pairs algorithm



Results

- Random subset of distances
- Approximate hierarchical clustering

7% of distances measured

- Final candidates of closest pairs algorithm
- Approximate hierarchical clustering

1% of distances measured

6000 datapoints, 80-dimensional

SAME QUALITY!

Problems

- The algorithm for finding the candidates for similarity is not very efficient – if the dimensionality is small then it may be faster to calculate all the distances.
- The algorithm needs $2N^2$ bytes of memory – 1.5 GB for about 25000 human genes.

Future

- Better approximation algorithm

Future

- Better approximation algorithm
- Optimise for speed

Future

- Better approximation algorithm
- Optimise for speed
- Parallelise