Hierarchical Clustering

TEAM YOLO
Zeqing Jin
Xianlin Shao
Yifei Zhang
Zilan Zhang

Berkeley
UNIVERSITY OF CALIFORNIA

Introduction

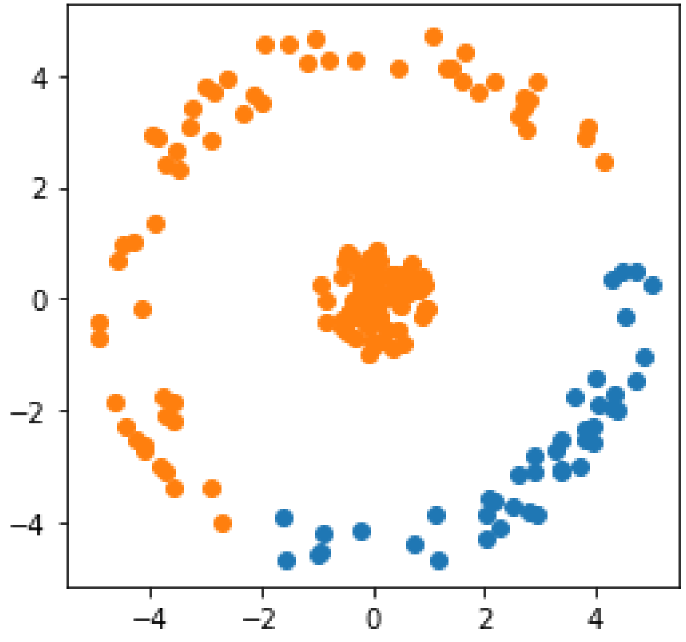
- Hierarchical clustering is a general family of clustering algorithms that build clusters by merging or splitting them successively. [2]

- Two common hierarchy algorithms:
  - Agglomerative clustering
  - Divisive clustering

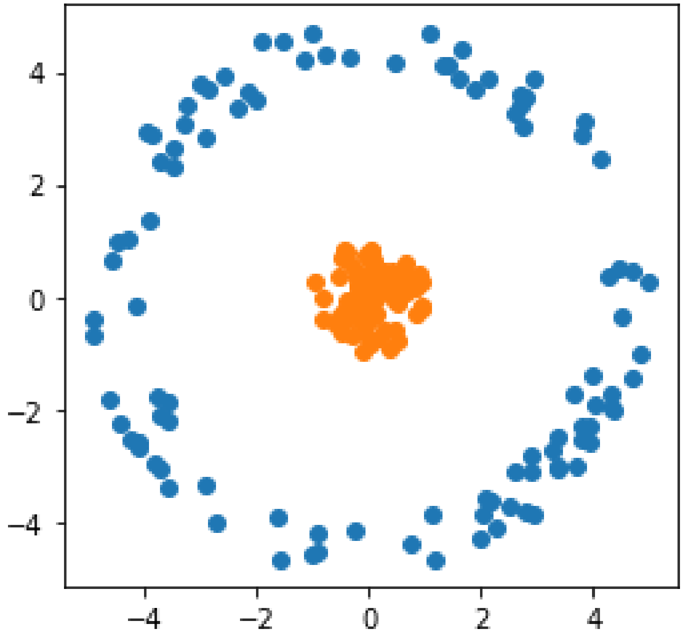
Berkeley
UNIVERSITY OF CALIFORNIA

Limitations of K-means Clustering

- Non-spherical data points
- Prior assumption of similar number of data points in each cluster



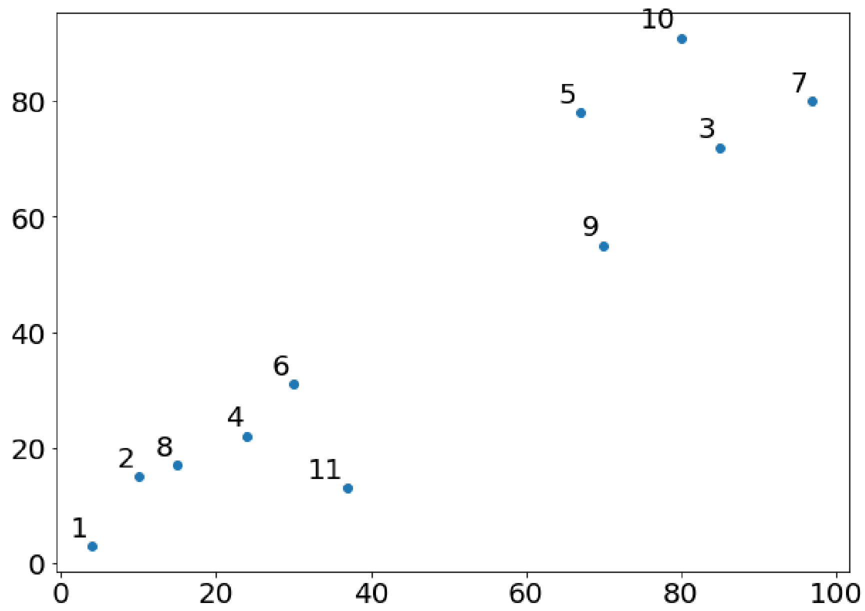
K-means

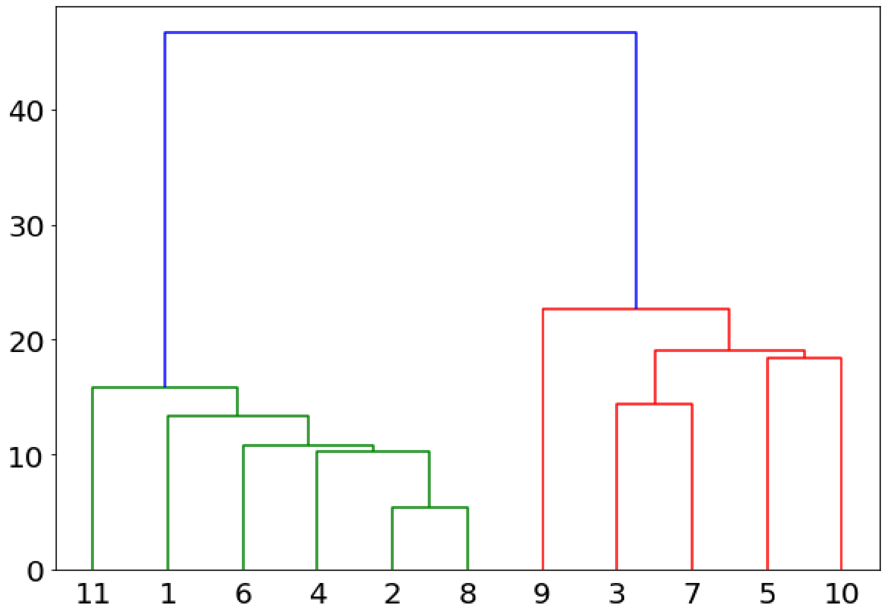


Hierarchical

Dendrogram

- Dendrogram is a tree-like hierarchy which shows the relationship between objects.

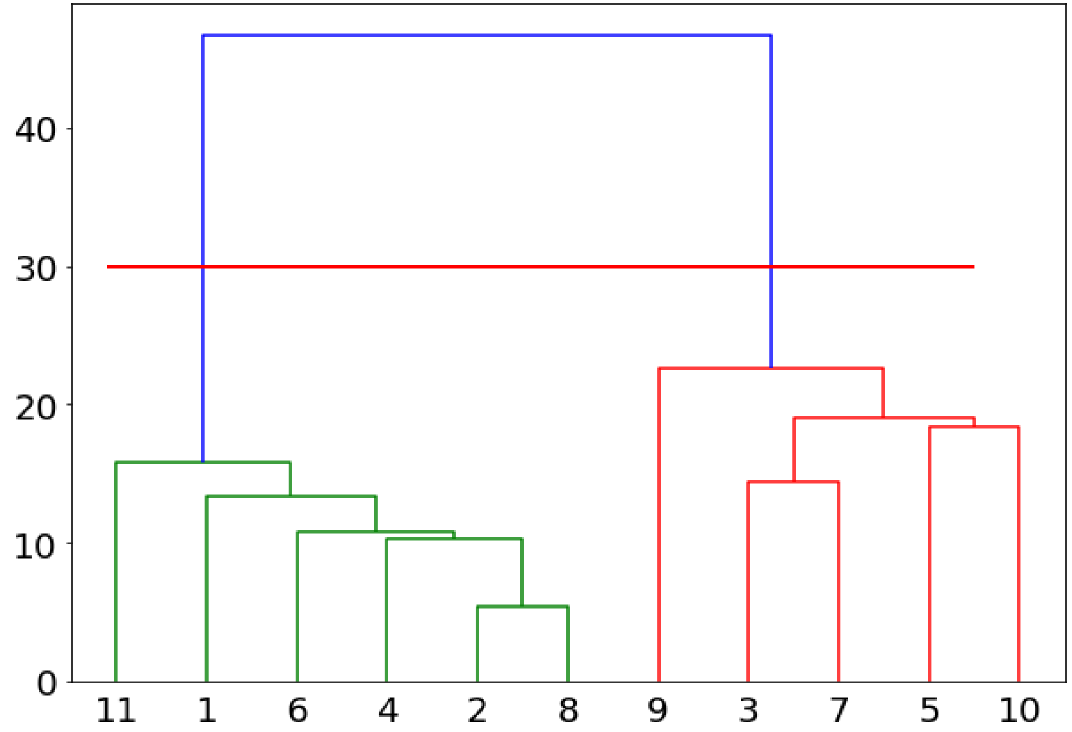




Berkeley
UNIVERSITY OF CALIFORNIA

Dendrogram

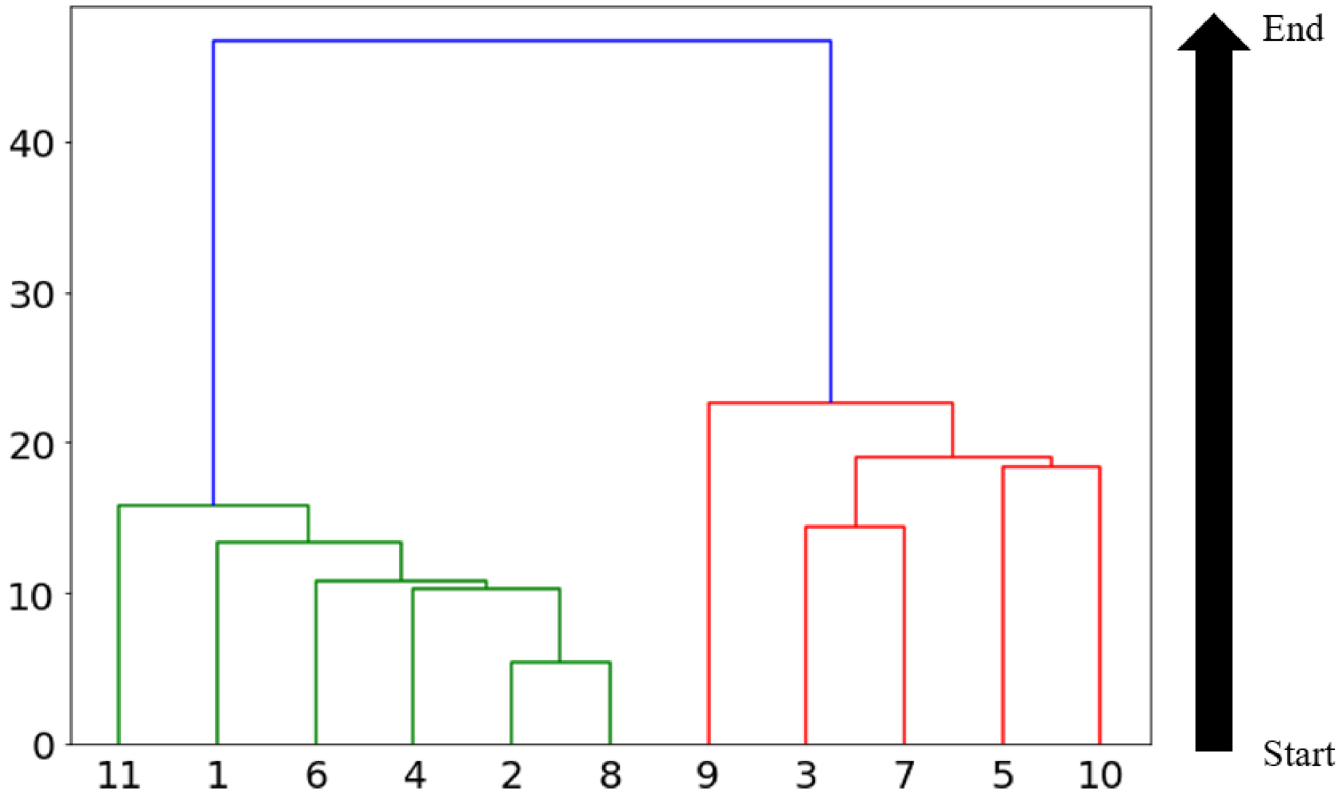
- Dendrogram implicitly contains all possible values of the number of clusters
- Shows relative relations between clusters (points)
- Fails to show all the absolute distances between points



Berkeley
UNIVERSITY OF CALIFORNIA

Agglomerative Clustering

- Start with n clusters containing one single point.
- End up with one cluster containing n objects.



Berkeley UNIVERSITY OF CALIFORNIA

Agglomerative Clustering [1]

Algorithm 1: Agglomerative Hierarchical Clustering

Input: n data points

Output: final clustering result over n data points

```
Initialize n clusters c_i, i = 1, ..., n;
Initialize the dissimilarity matrix;
for the number of clusters k decreases from n to 1 do
    Find the two clusters c_i, c_j with the smallest dissimilarity according to dissimilarity matrix;
    Merge c_i with c_j and update the dissimilarity matrix;
end for
```

Agglomerative Clustering

- Euclidean distance between points:

$$d(i,j) = \sqrt{\Sigma_{p-1}^{q}\left(x_{ip} - x_{jp}\right)^2}$$

  where q is dimension of the point

- Dissimilarity matrix:

$$\boldsymbol{S} = \begin{bmatrix} d(1,1) & \cdots & d(1,n) \\ \vdots & \ddots & \vdots \\ d(n,1) & \cdots & d(n,n) \end{bmatrix}$$

Berkeley
UNIVERSITY OF CALIFORNIA

Distance between clusters

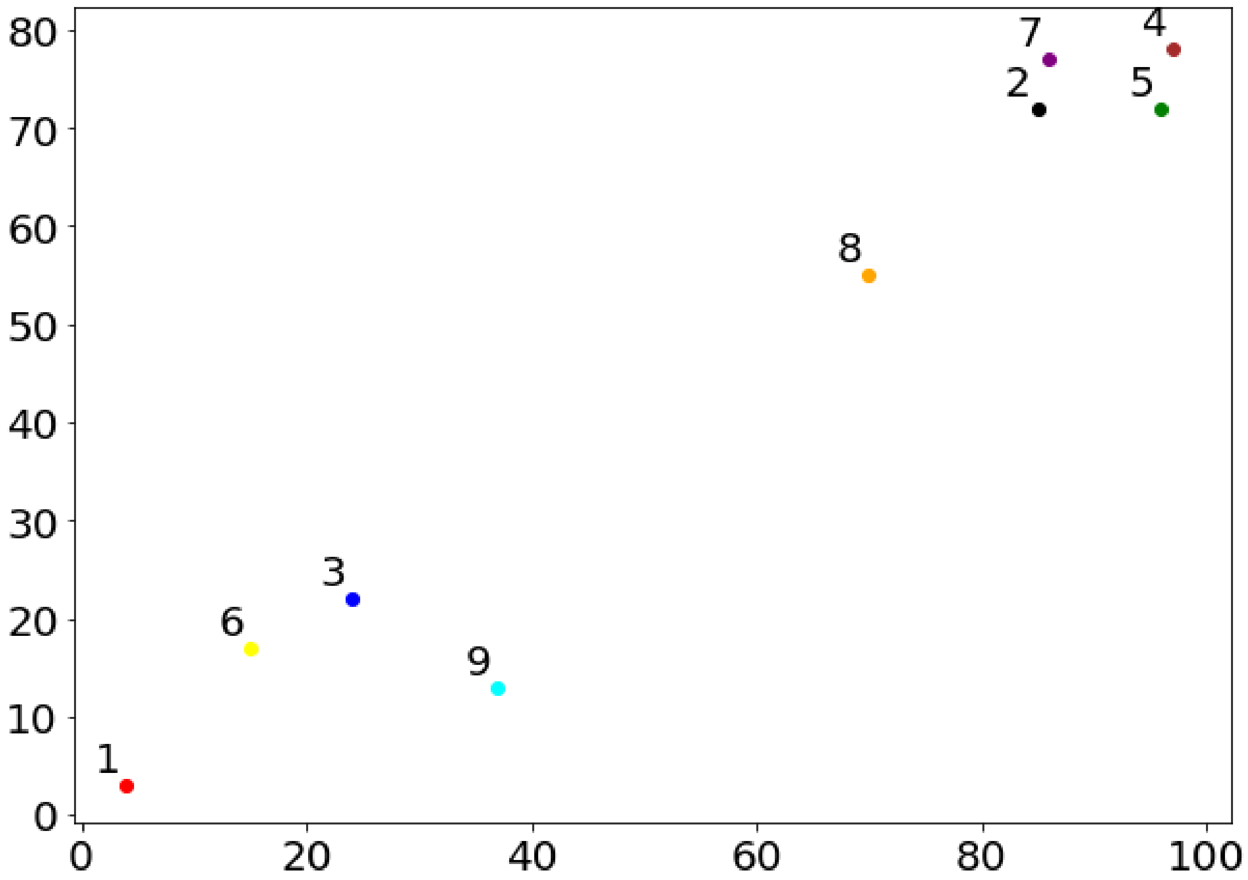
- **Complete linkage:**
  - Maximum distance between clusters
- **Single linkage:**
  - Minimum distance between clusters
- **Average linkage:**
  - Average distance between clusters
- **Centroid linkage:**
  - Distance between centroids of clusters
- **Ward’s linkage:**
  - Increase in sum of squares if two clusters are merged

Berkeley
UNIVERSITY OF CALIFORNIA

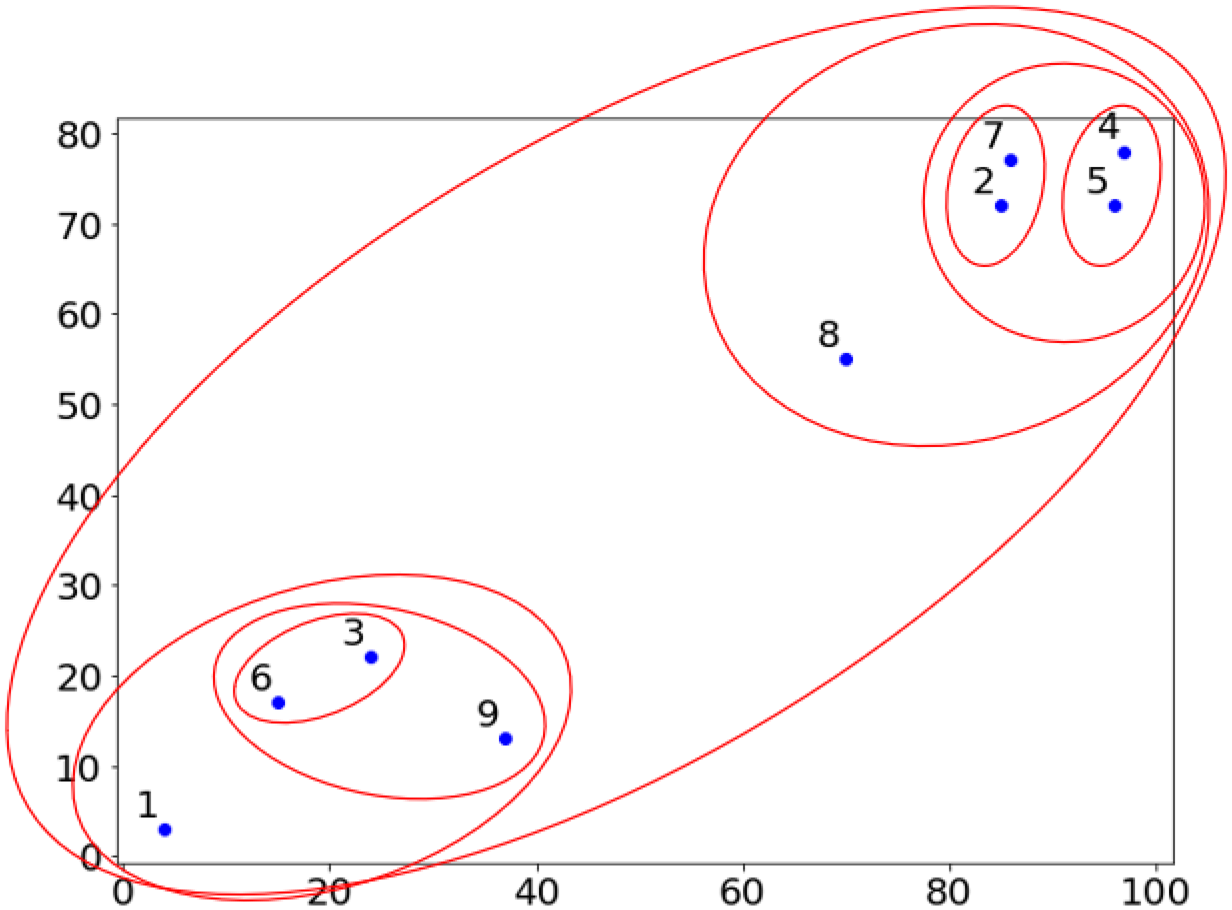
Agglomerative Clustering Example

- Start from 9 clusters
- Complete linkage
- 9×9 dissimilarity matrix

How many distances do we need to calculate? 81?

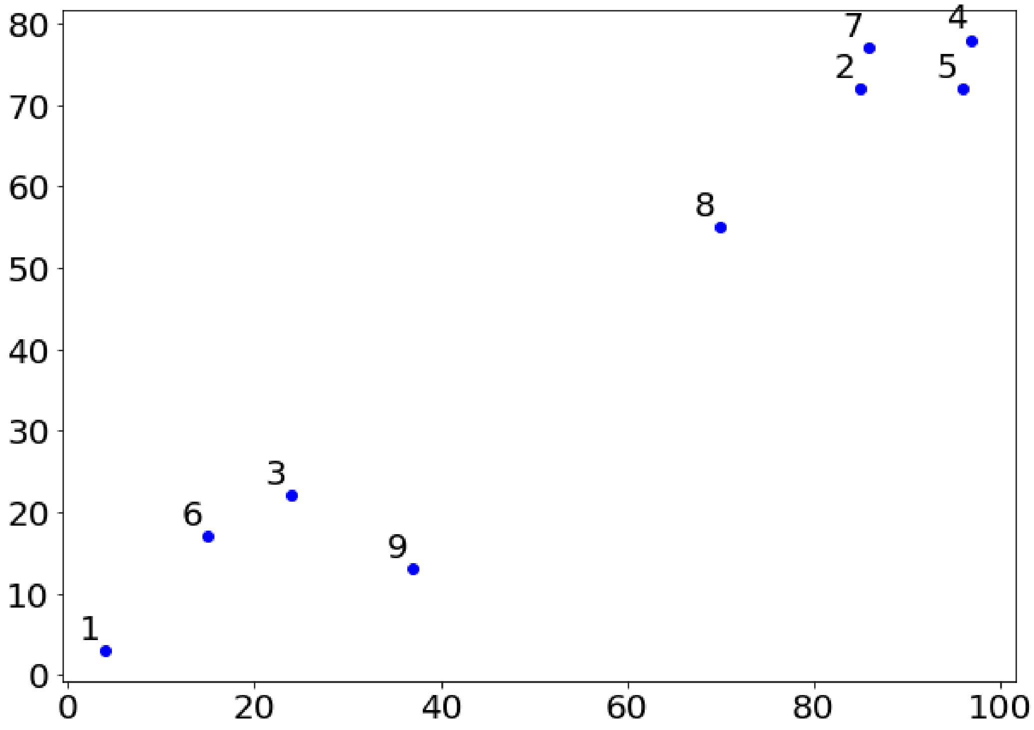


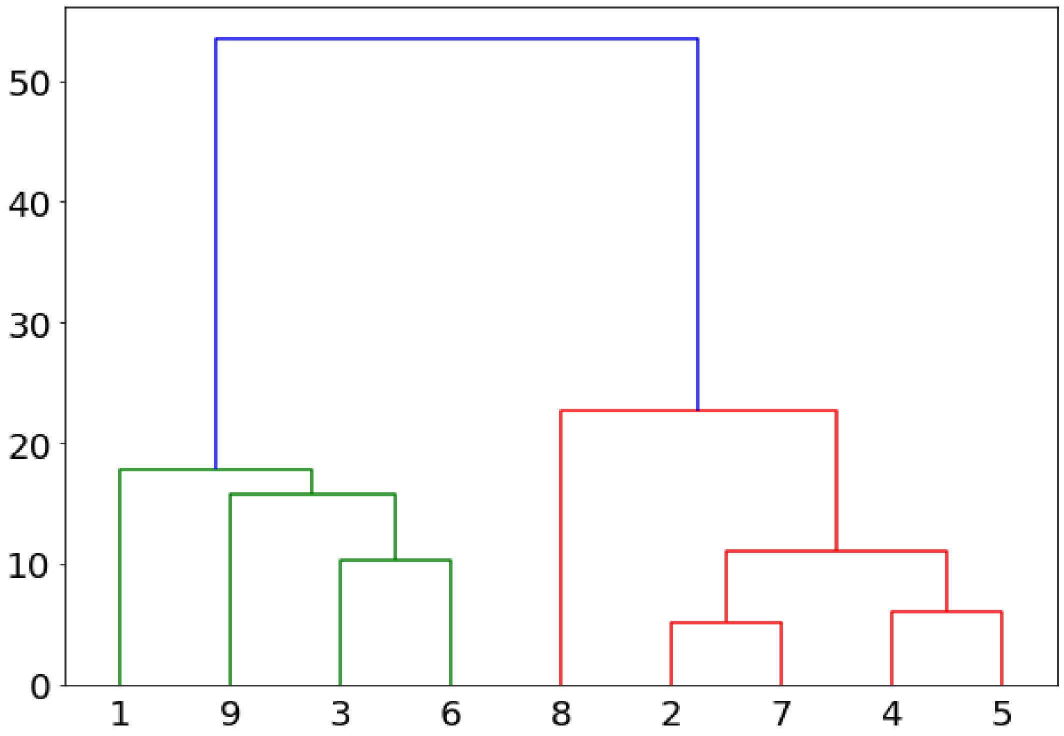
Agglomerative Clustering Example



Berkeley
UNIVERSITY OF CALIFORNIA

Agglomerative Clustering Example

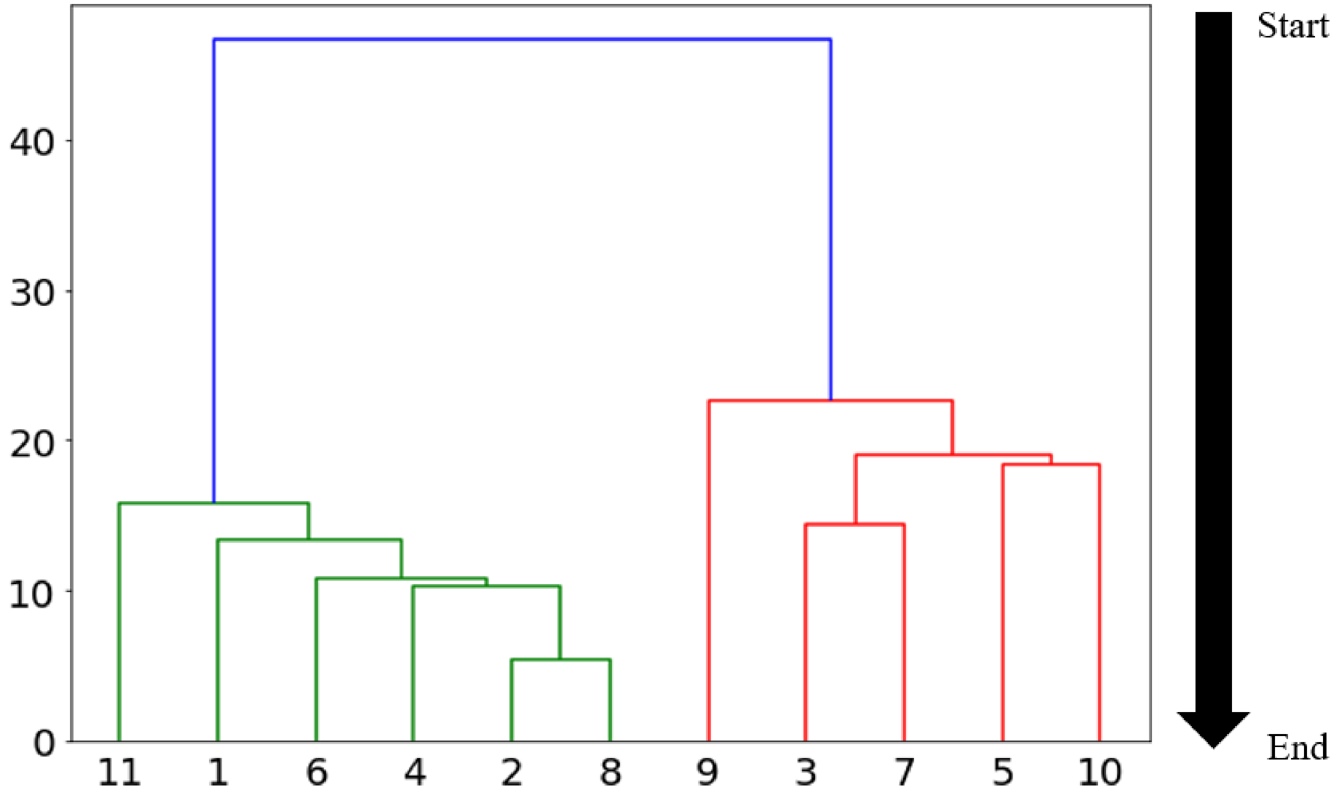




Berkeley
UNIVERSITY OF CALIFORNIA

Divisive Clustering (DIANA)

- Start with one cluster containing all n points.
- End up with n clusters containing one object.



Divisive Clustering (DIANA) [1]

Algorithm 2: Divisive Analysis Clustering (DIANA)

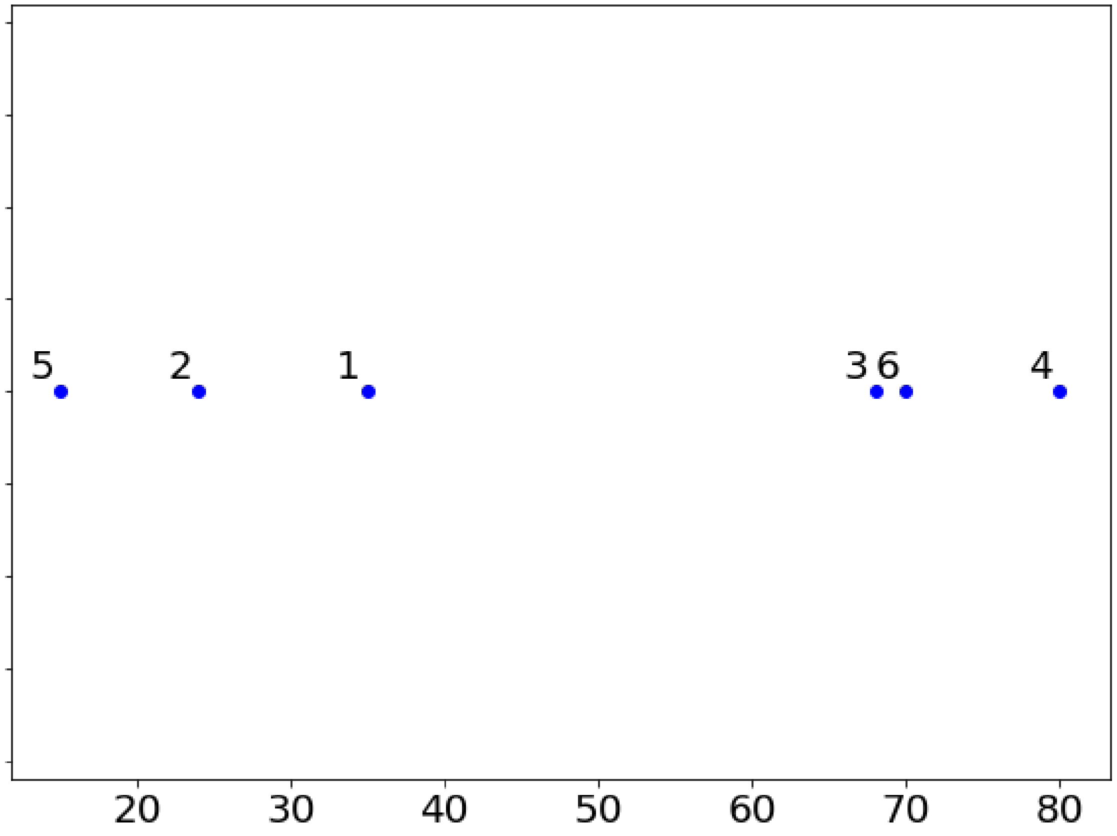
```
Input: n data points
Output: final clustering result
Initialize one cluster with all objects c_1;
for the number of clusters k increases from 1 to n do
    Choose the cluster C_i with the largest diameter value;
    Within C_i, choose the object that has the maximum distance with the other objects as one
      cluster and split this object as a splinter cluster;
    Update C_i;
    while True do
        for each data point j in C_i do
            Calculate the distance d_1 between the data j and the other objects in C_i as one
              cluster;
            Calculate the distance d_2 between the data j and the splinter cluster;
            Calculate the difference δd_j = d_1 − d_2;
        end for
        if max δd_j is positive then
            Move the data j with positive δd_j to the splinter cluster and update C_i;
        else
            break;
        end if
    end while
end for
```

Berkeley
UNIVERSITY OF CALIFORNIA

Divisive Clustering Example

- Start from one cluster
- Complete linkage
- 6×6 dissimilarity matrix

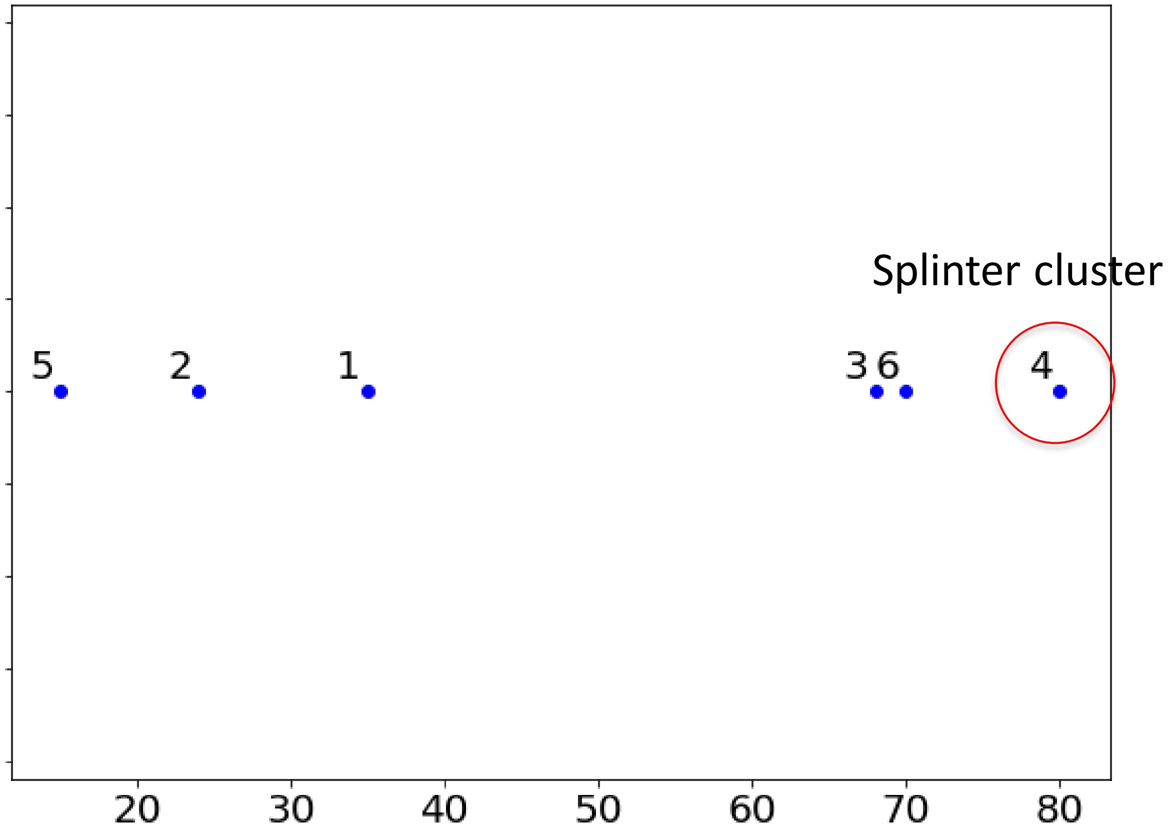
$$S = \begin{bmatrix} 0 & 11 & 33 & 45 & 20 & 35 \\ 11 & 0 & 44 & 56 & 9 & 46 \\ 33 & 44 & 0 & 12 & 53 & 2 \\ 45 & 56 & 12 & 0 & 65 & 10 \\ 20 & 9 & 53 & 65 & 0 & 55 \\ 35 & 46 & 2 & 10 & 55 & 0 \end{bmatrix}$$



Divisive Clustering Example

- Calculate the distance between each point and the other objects

Point	Distance to other points
1	45
2	56
3	53
4	65
5	65
6	55



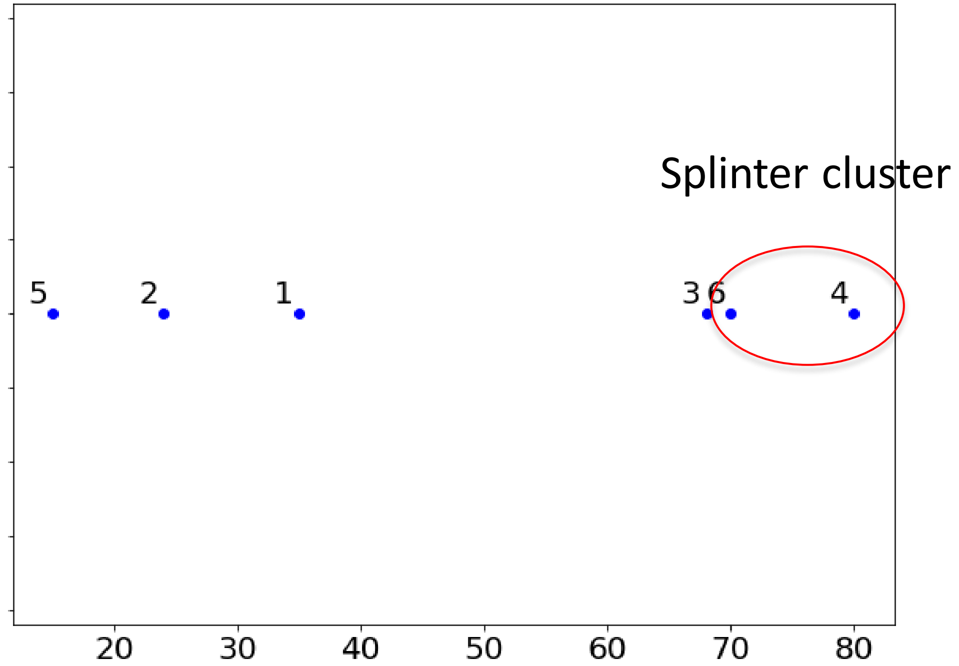
- Splinter cluster {4}

Divisive Clustering Example

- Calculate the distance between each remaining point and the other objects

- Also the distance to the splinter cluster

- Splinter cluster {4,6}

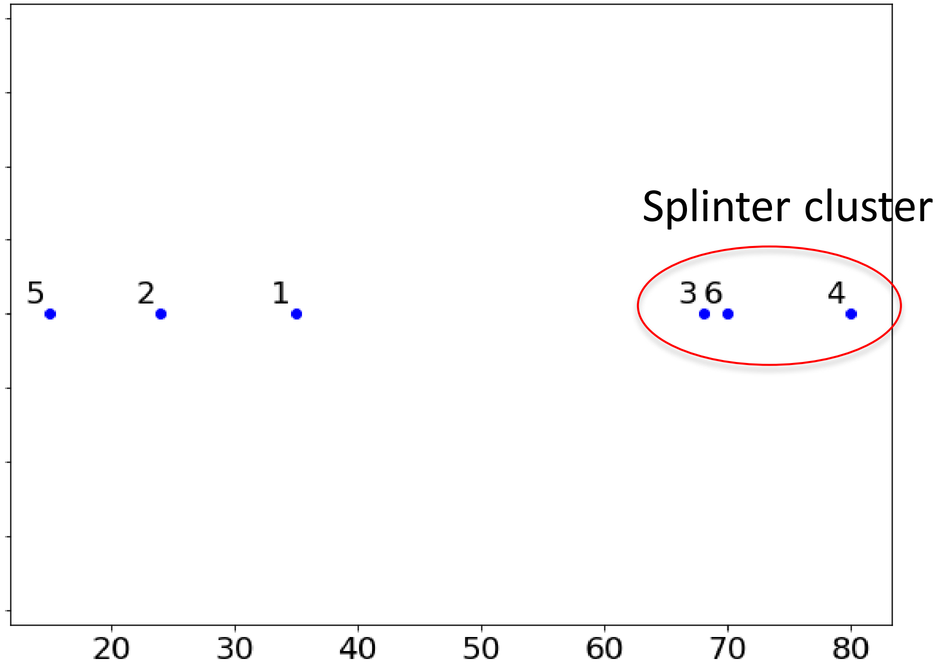


Point	Distance to other points	Distance to the splinter cluster	Difference
1	35	45	-10
2	46	56	-10
3	53	12	41
5	55	65	-10
6	55	10	45

Berkeley
UNIVERSITY OF CALIFORNIA

Divisive Clustering Example

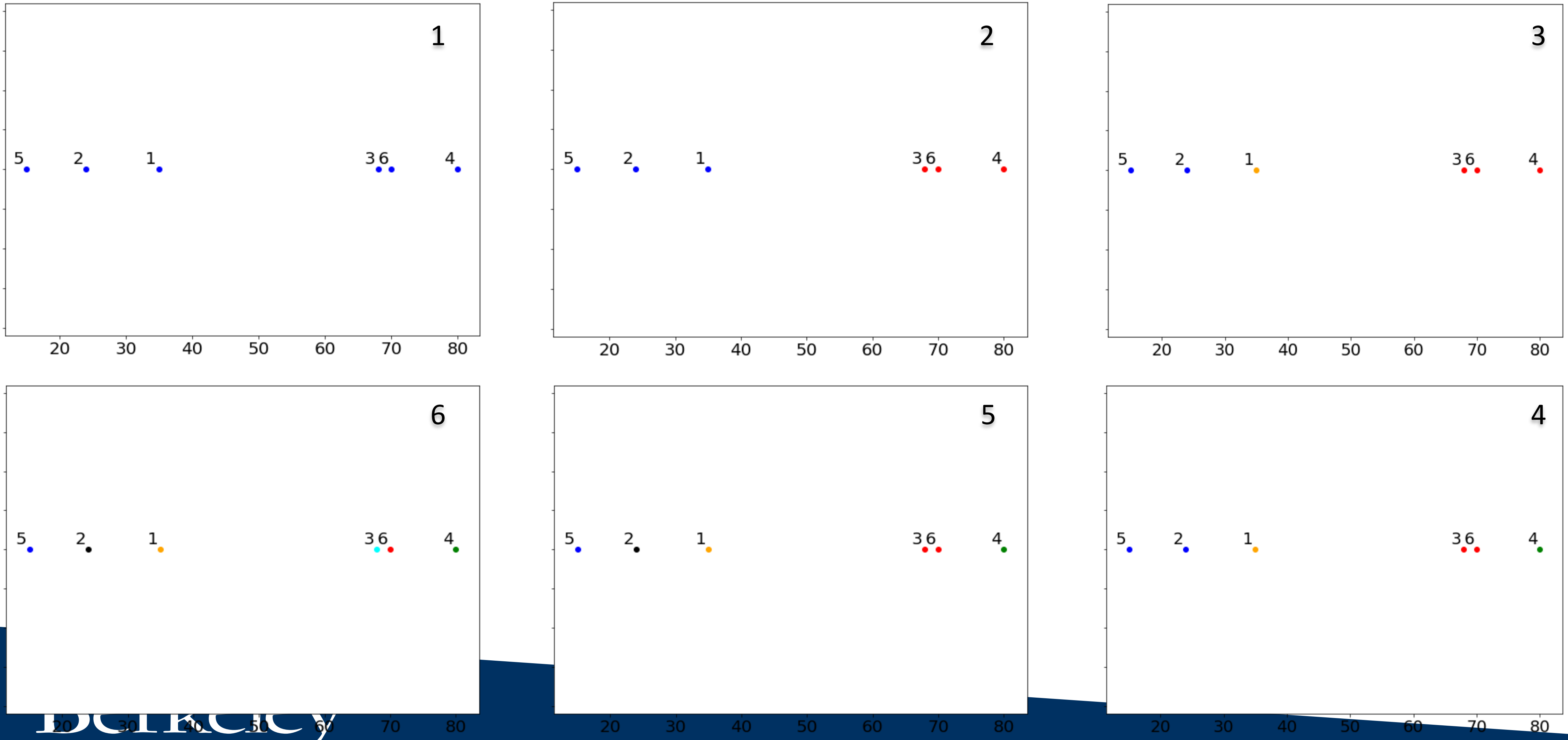
- Repeat the previous step
- Splinter cluster {4,6,3}
- {1,2,3,4,5,6} into {1,2,5} and {4,6,3}



Point	Distance to other points	Distance to the splinter cluster	Difference
1	33	45	-12
2	44	56	-12
3	53	12	41
5	53	65	-12

Berkeley
UNIVERSITY OF CALIFORNIA

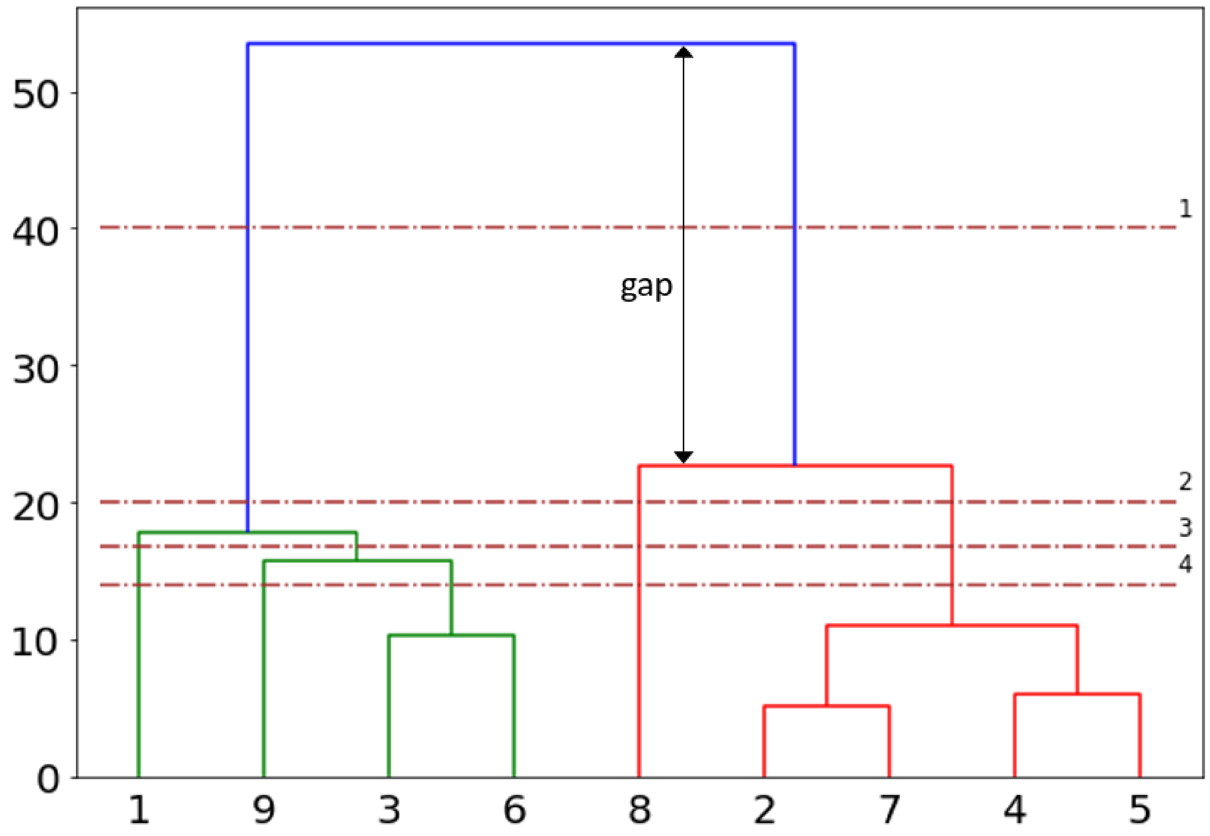
Divisive Clustering (DIANA)



Berkeley
UNIVERSITY OF CALIFORNIA

Determination of k

- (General) Elbow method [2]
  - Total within-cluster sum of square (WSS)
- (Dendrogram) Cut at different dissimilarity levels gives multiple values of k
- Cut at the largest dissimilarity gap gives a roughly reasonable k
- Affected by the linkage type since dissimilarity may change after each iteration.



Berkeley UNIVERSITY OF CALIFORNIA

Specific Hierarchical Algorithms

- Linkage algorithm
  - Single linkage, average linkage, complete linkage

- CURE (Clustering Using REpresentatives)

- BIRCH (Balanced Iterative Reducing and Clustering using Hierarchies) (Optional)

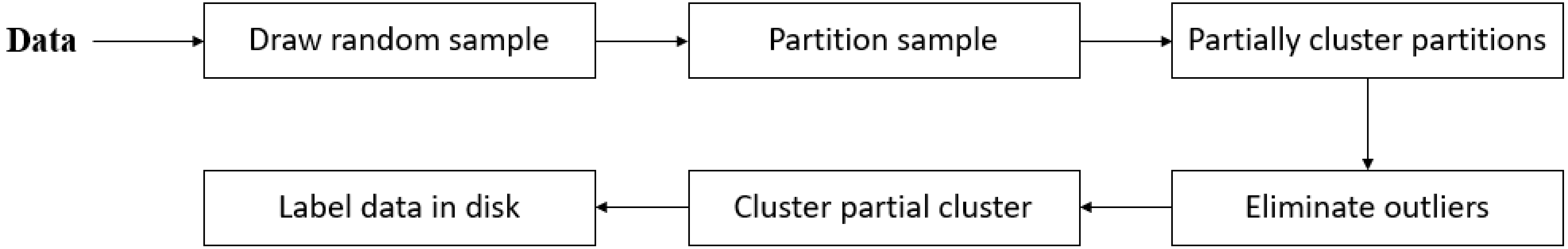
Linkage algorithm [4]

- Single linkage:
  - Time complexity $O[n^3]$ (simplest implementation)
  - Sensitive to outliers
- Complete linkage
  - Time complexity can be reduced to $O[n^2 \log n]$
  - Cluster similar objects
- Average linkage
  - Compromise between single and complete
  - Often fails in complicated cluster shapes

Berkeley
UNIVERSITY OF CALIFORNIA

CURE (Clustering Using REpresentatives) [4]

- A hierarchical based clustering technique



- Representative points and shrinking factor
- Apply to outliers

Berkeley
UNIVERSITY OF CALIFORNIA

Reference

- [1]. Leonard Kaufman and Peter J Rousseeuw. *Finding groups in data: an introduction to cluster analysis*. Vol. 344. John Wiley & Sons, 2009.
- [2]. Bradley Boehmke Brandon Greenwell. *Hands-On Machine Learning with R*. Feb. 2020. URL: https : / / bradleyboehmke . github . io / HOML / hierarchical . html # fig:dendrogram2.
- [3]. Godfrey and Kate. *Determining The Optimal Number Of Clusters: 3 Must Know Methods*. Feb. 2020. URL: https://bradleyboehmke.github.io/HOML/kmeans.html#eq:tot-within-ss.
- [4]. M Kuchaki Rafsanjani, Z Asghari Varzaneh, and N Emami Chukanlo. “A survey of hierarchical clustering algorithms”. In: *The Journal of Mathematics and Computer Science* 5.3 (2012), pp. 229–240.

Berkeley
UNIVERSITY OF CALIFORNIA