* Hierarchical Clustering *

↳ Unsupervised ML Algo, group unlabelled dataset into a cluster c/a Hierarchical Cluster Analysis.

↳ Develop tree-structured hierarchy of clusters, tree-shaped structure c/a dendrogram.

↳ Num. of clusters not predefined.

↳ 2 approaches:-

① Agglomerative: Bottom-up approach.

↳ All data points consider individual clusters & iteratively merge until 1 cluster left.

② Divisive: Top-down approach.

- Reverse of Agglomerative.
- Start with consider 1 cluster and iteratively divide till reach individual cluster.

↳ Working of Hierarchical Clustering:-

① Start, consider every data point separate individual cluster.

② Identify 2 closest together clusters.

③ Merge the clusters.

④ Continue until all clusters merged together.

1) Agglomerative Clustering :- Bottom-up method.

① consider every data point individual cluster.

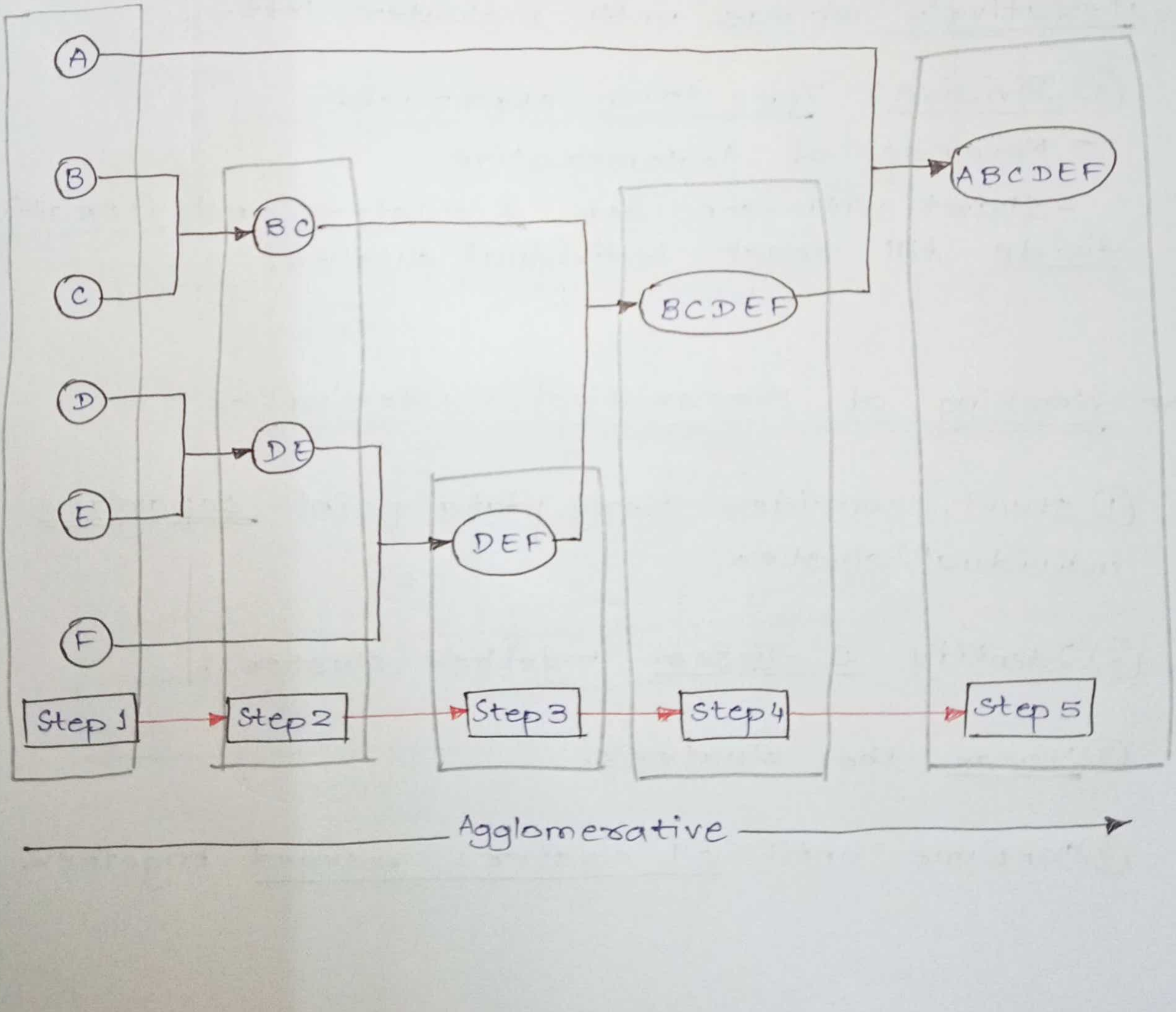
② calculate proximity matrix (similarity of cluster with all other clusters).

③ Merge highly similar / closest clusters.

④ Recalculate proximity matrix of each clusters.

⑤ Repeat steps 3 & 4, until only 1 cluster remain.

- Example :



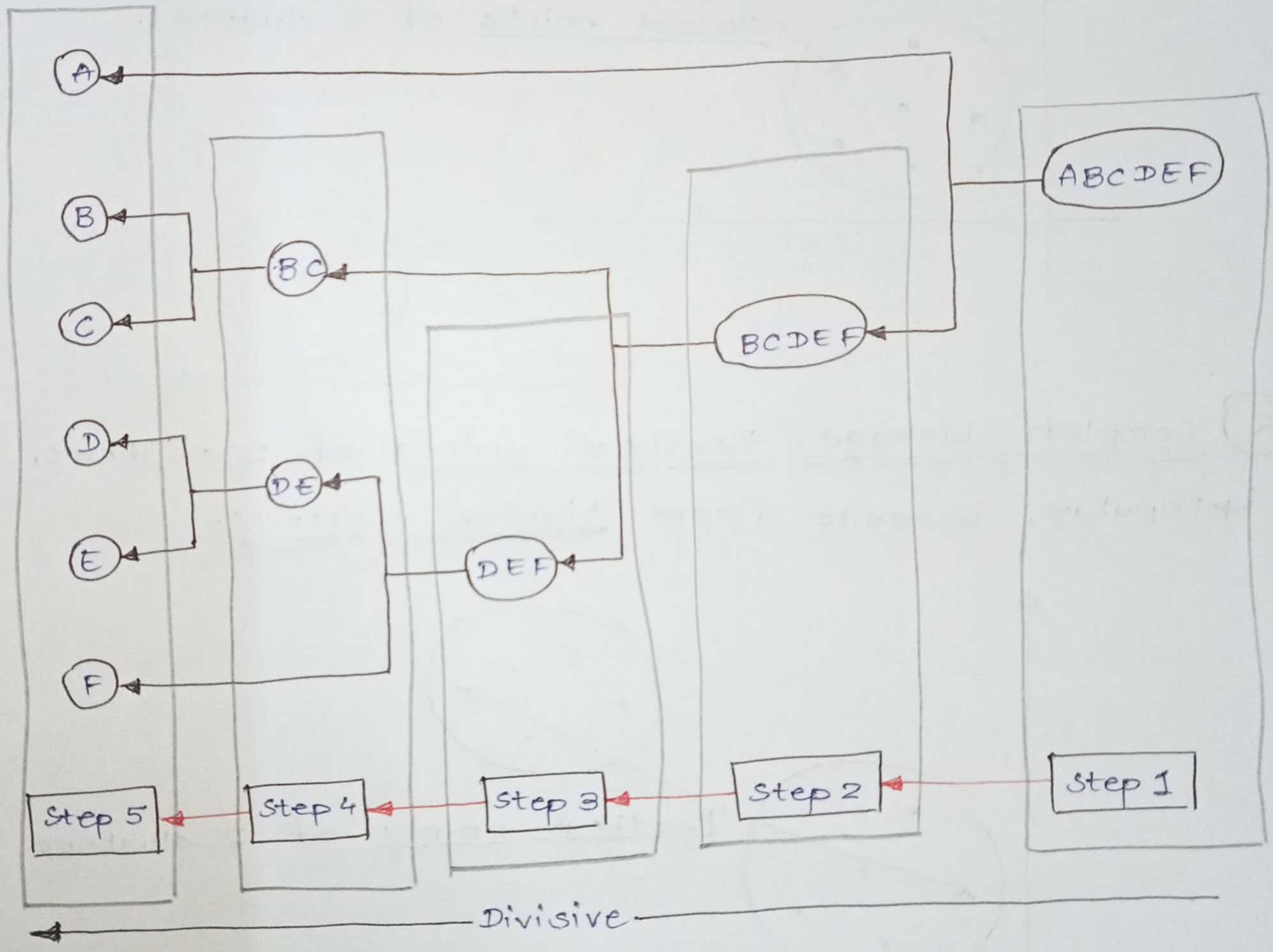
2) Divisive Clustering:- Top-down method.

↳ Consider all data points a single cluster.

↳ In each iteration, separate data points from clusters which aren't comparable.

↳ At the end get N-clusters.

- Example -

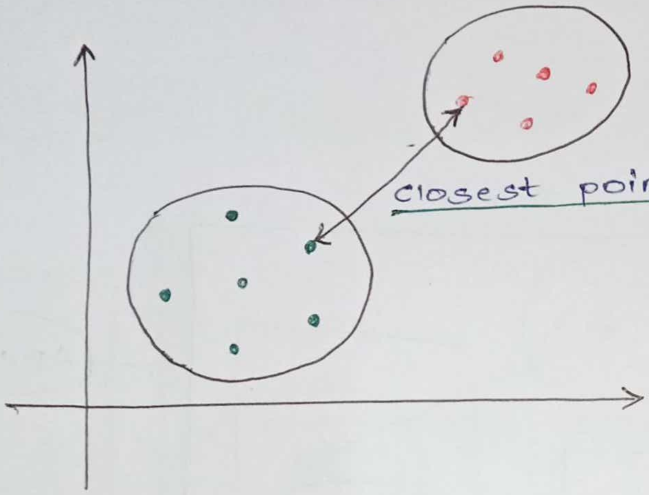


- Measure Dist. bet^n 2 clusters :-

↳ Find closest distance bet^2 2 clusters.

↳ Different methods c/a Linkage Methods.

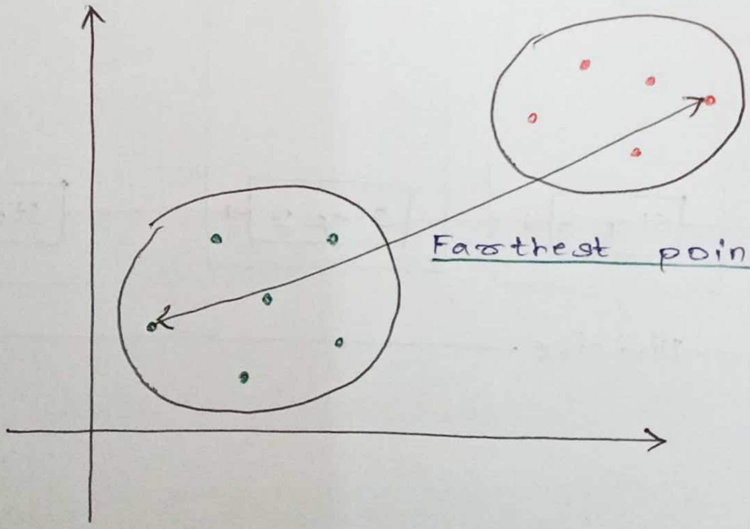
① Single Linkage : shortest distance between the closest points.



closest points of 2 clusters.

② Complete Linkage : Farthest points of 2 clusters.

↳ Popular, because form tighter clusters.



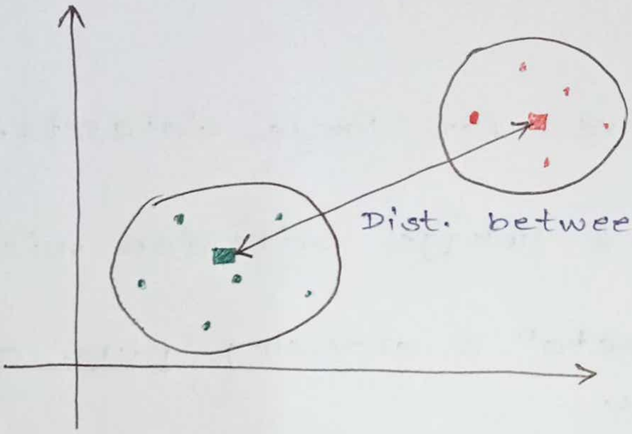
Farthest points of 2 clusters.

③ **Average Linkage**:- Calculate average distances.

↳ Add up dist. betn each pair of datasets & then divide by total num. of data points.

$$\left(\begin{matrix}\text{Average}\\\text{Dist}\end{matrix}\right) = \frac{\sum \text{dist. bet}^n\text{ each data pair}}{\text{Num. of data points}}$$

④ **Centroid Linkage**: Dist. betn centroids.



- Advantages:

↳ Not required to predefine num. of clusters.

↳ Hierarchical cluster representation useful for multiple level data points.

↳ Work well for different dist. metrics & linkage criterion.

- Limitations:

↳ Computationally expensive for large datasets.

↳ Choose dist. metric & linkage criterion wisely.

↳ Dendrogram interpretaⁿ & choosing num. of clusters affect results.