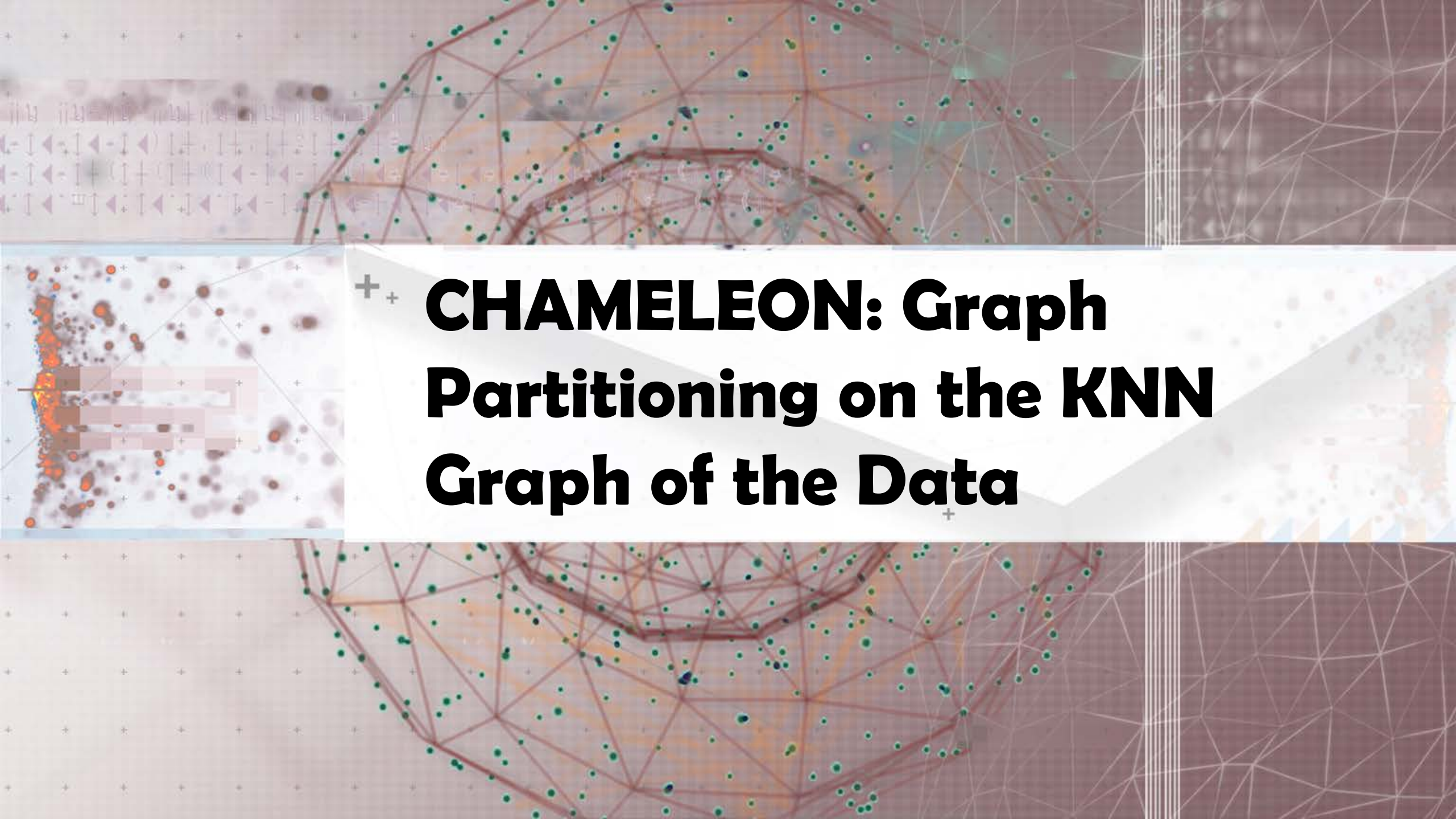
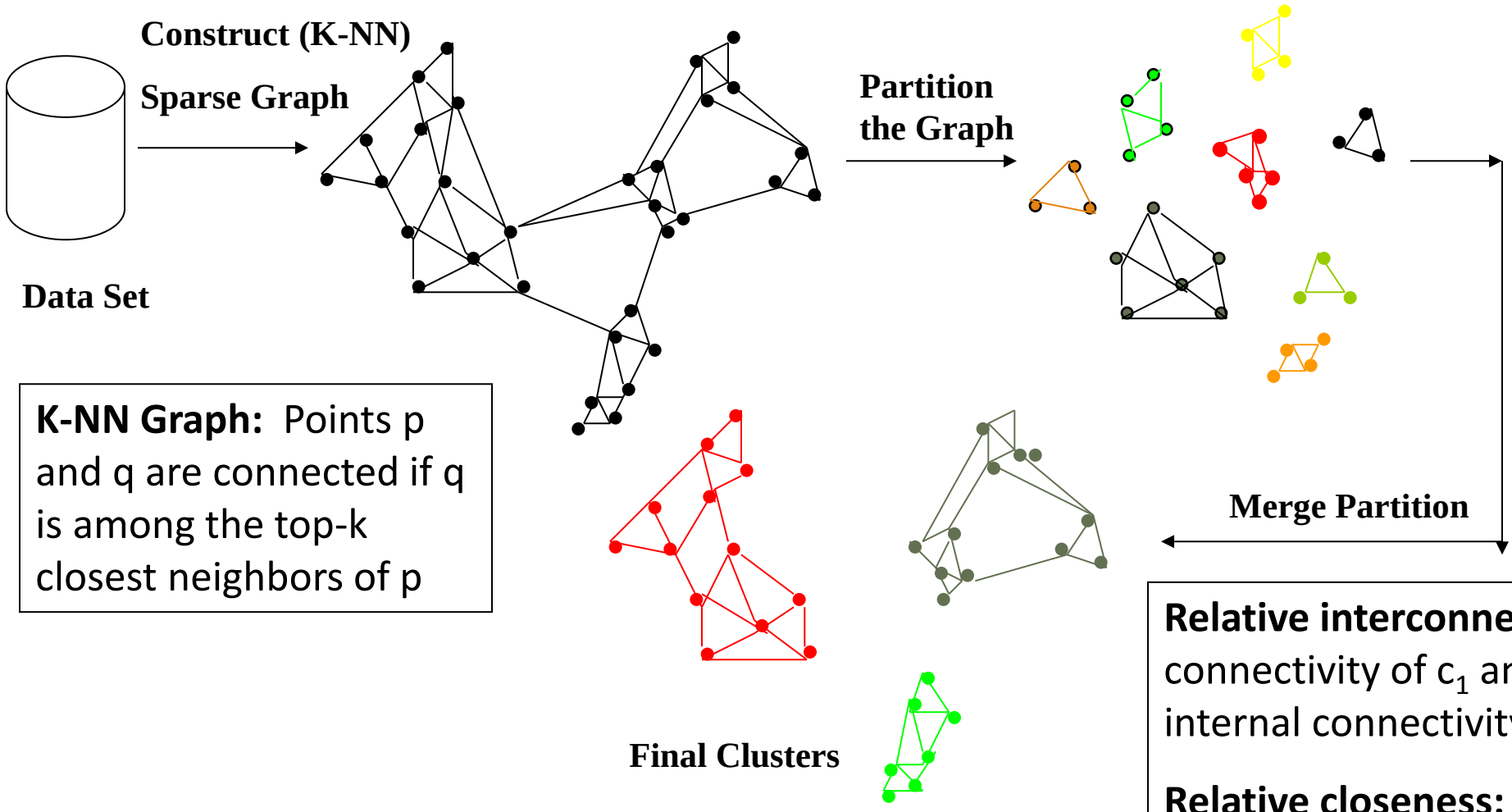


# CHAMELEON: Graph Partitioning on the KNN Graph of the Data

# CHAMELEON: Hierarchical Clustering Using Dynamic Modeling

- CHAMELEON: A graph partitioning approach (G. Karypis, E. H. Han, and V. Kumar, 1999)
- Measures the similarity based on a dynamic model
  - Two clusters are merged only if the *interconnectivity* and *closeness (proximity)* between two clusters are high *relative to* the internal interconnectivity of the clusters and closeness of items within the clusters
- A graph-based, two-phase algorithm
  1. Use a graph-partitioning algorithm: Cluster objects into a large number of relatively small sub-clusters
  2. Use an agglomerative hierarchical clustering algorithm: Find the genuine clusters by repeatedly combining these sub-clusters

# Overall Framework of CHAMELEON



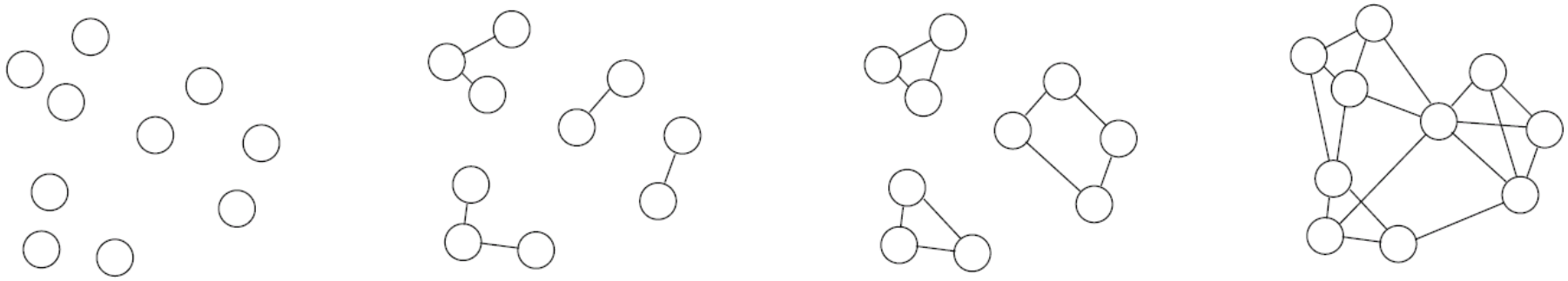

**K-NN Graph:** Points p and q are connected if q is among the top-k closest neighbors of p

**Relative interconnectivity:** connectivity of $c_1$ and $c_2$ over internal connectivity

**Relative closeness:** closeness of $c_1$ and $c_2$ over internal closeness

# KNN Graphs and Interconnectivity

- K-nearest neighbor (KNN) graphs from an original data in 2D:



(a) Original Data in 2D (b) 1-nearest neighbor graph (c) 2-nearest neighbor graph (d) 3-nearest neighbor graph

- $EC_{\{Ci,Cj\}}$ :The absolute interconnectivity between $C_i$ *and* $C_j$*:*
  - *The sum* of the weight of the edges that connect vertices in $C_i$ *to vertices in* $C_j$
- Internal interconnectivity of a cluster $C_i$ : *The size of its min-cut bisector* $EC_{Ci}$ *(i.e., the weighted sum* of edges that partition the graph into two roughly equal parts)
- Relative Interconnectivity (RI):

$$RI(C_i, C_j) = \frac{|EC_{\{C_i,C_j\}}|}{\frac{|EC_{C_i}|+|EC_{C_j}|}{2}}$$

# Relative Closeness & Merge of Sub-Clusters

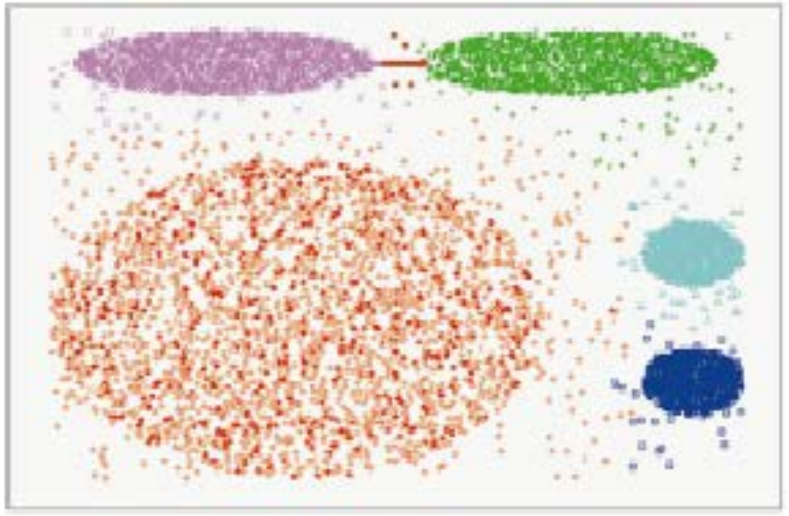
- **Relative closeness** between a pair of clusters $C_i$ *and* $C_j$ *: The absolute closeness* between $C_i$ *and* $C_j$ *normalized w.r.t. the internal closeness of the two clusters* $C_i$ *and* $C_j$

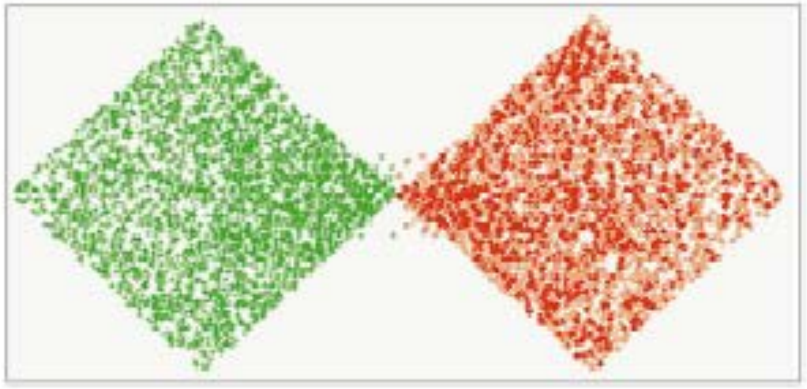
$$RC(C_i, C_j) = \frac{\overline{S}_{EC_{\{C_i, C_j\}}}}{\frac{|C_i|}{|C_i|+|C_j|}\overline{S}_{EC_{C_i}} + \frac{|C_j|}{|C_i|+|C_j|}\overline{S}_{EC_{C_j}}}$$

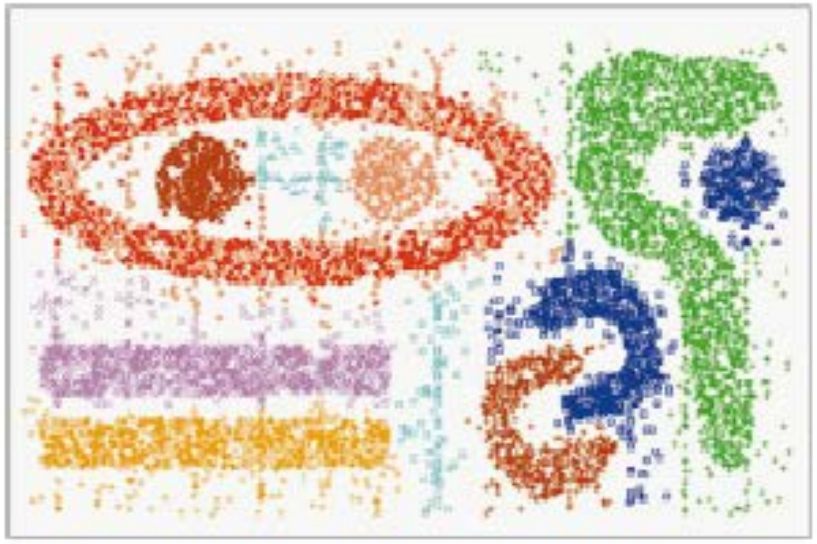
  - where $\overline{S}_{EC_{C_i}}$ and $\overline{S}_{EC_{C_j}}$ are the average weights of the edges that belong to the min-cut bisector of clusters $C_i$ *and* $C_j$ *,* respectively, and $\overline{S}_{EC_{\{C_i, C_j\}}}$ is the average weight of the edges that connect vertices in $C_i$ *to vertices in* $C_j$

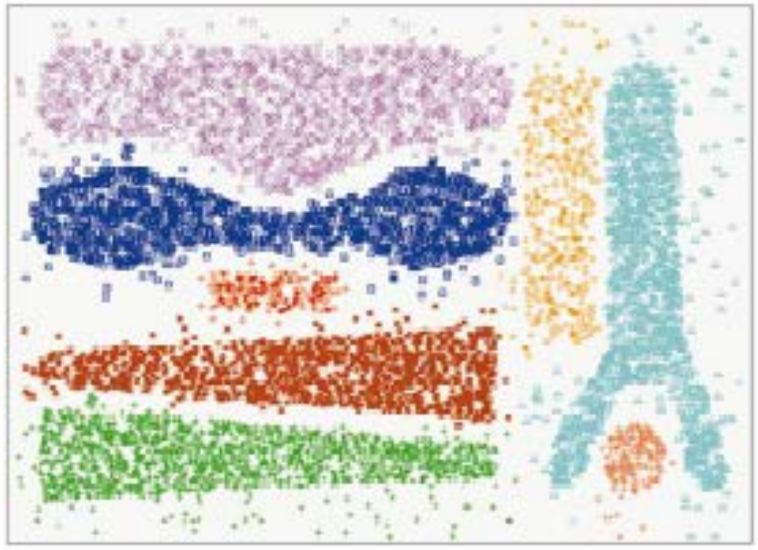
- **Merge Sub-Clusters:**
  - Merges only those pairs of clusters whose RI and RC are both above some user-specified thresholds
  - Merge those maximizing the function that combines RI and RC

# CHAMELEON: Clustering Complex Objects









CHAMELEON is capable to generate quality clusters at clustering complex objects