Density-Based Clustering Methods

- **Clustering based on density (local cluster criterion), such as density-connected points**

- **Major features:**
  - Discover clusters of arbitrary shape
  - Handle noise
  - Need density parameters as termination condition
- **Several interesting studies:**
  - **DBSCAN: Ester, et al**
  - OPTICS: Ankerst, et al
  - DENCLUE: Hinneburg & D. Keim
  - CLIQUE: Agrawal, et al.

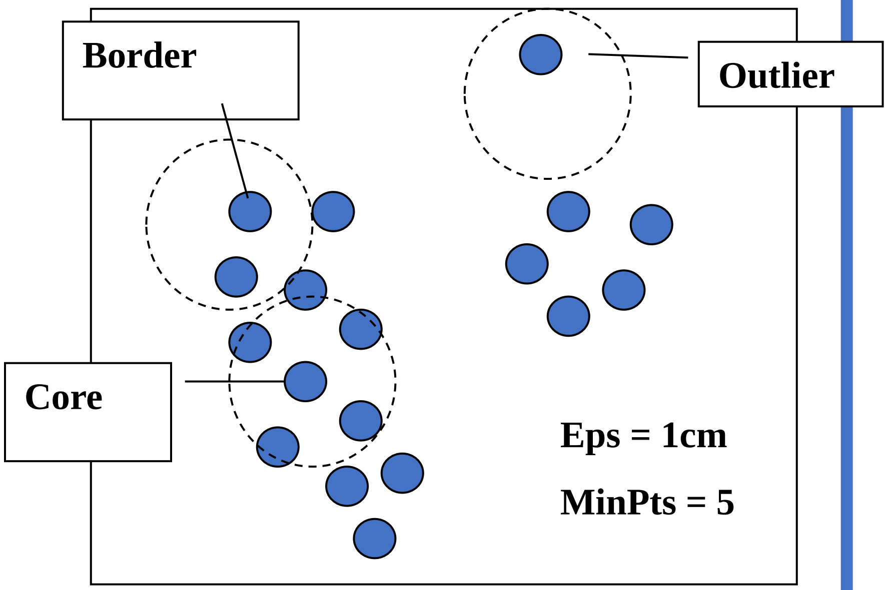
DBSCAN

Density Based Spatial Clustering of Applications with Noise

- Locates regions of high density separated by regions of low density.
- **Density** of a point is the number of points within the specified radius, **Eps**, of that point
- A ***cluster*** is defined as a maximal set of density-connected points

In center-based approach, we can classify point as being

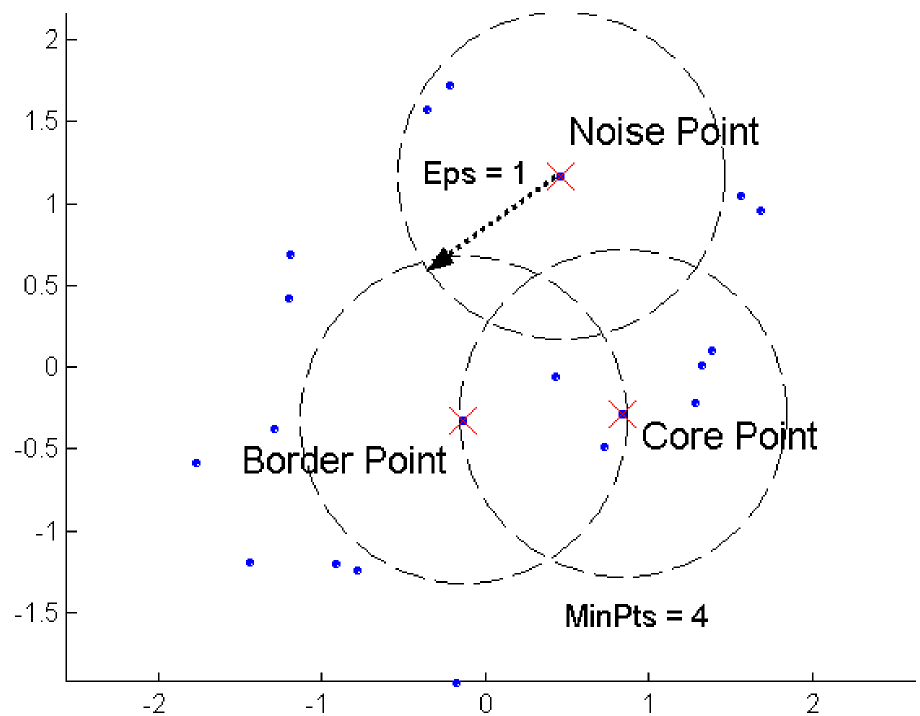
1. in the interior of dense region (**core**)
2. on the edge of a dense region (**border**)
3. in a sparely occupied region (**noise**)



DBSCAN

- DBSCAN is a density-based algorithm.
  - Density = number of points within a specified radius (Eps)

- A point is a core point if it has more than a specified number of points (MinPts) within Eps
- A border point has fewer than MinPts within Eps, but is in the neighborhood of a core point
- A noise point is any point that is not a core point or a border point.



DBSCAN

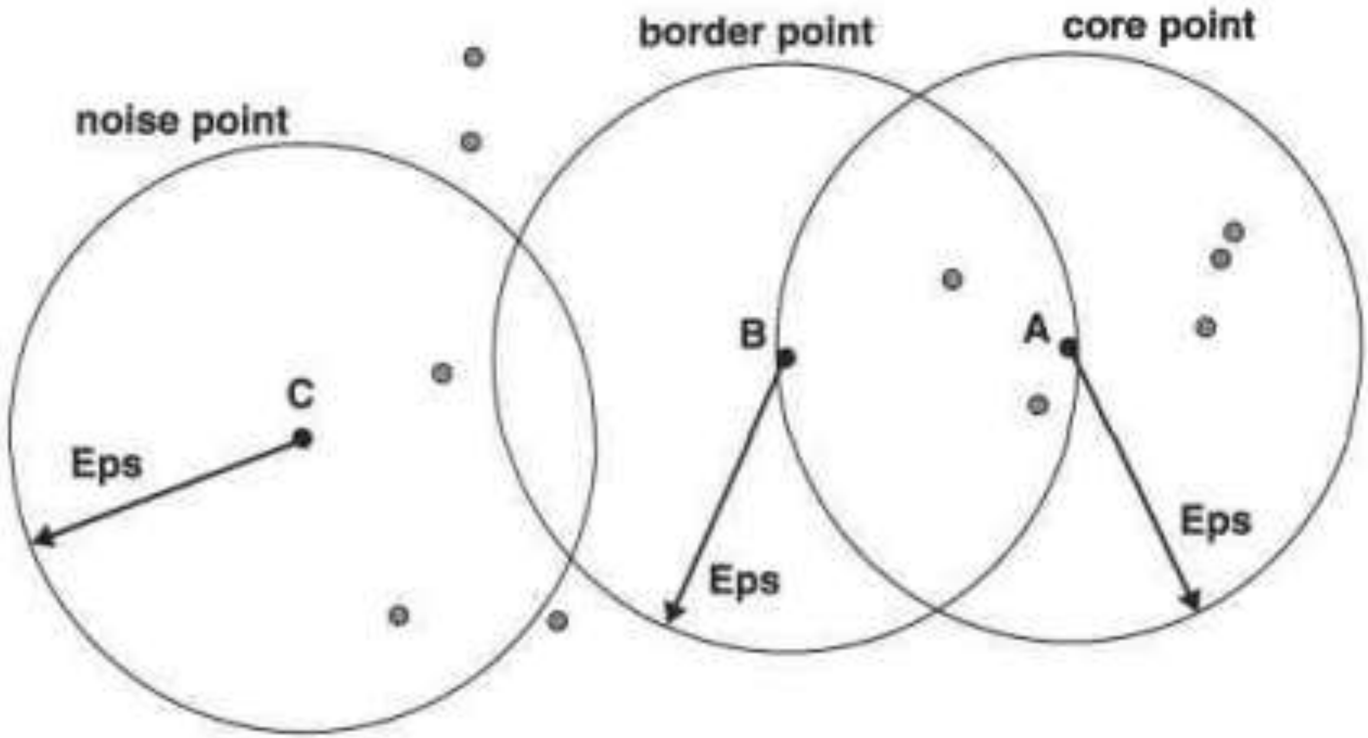


Figure 8.21. Core, border, and noise points.

DBSCAN: The Algorithm

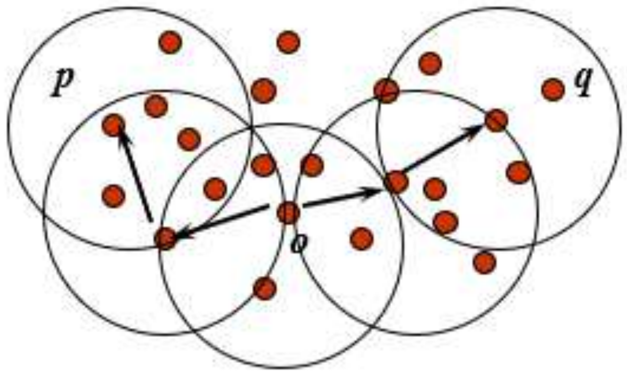
1. Label all points as core, border or noise.

2. Eliminate noise points.

3. Put an edge between all core points that are within Eps of each other.

4. Make each group of connected core points into a separate cluster.

5. Assign each border points to one of the clusters of its associated core points.



DBSCAN -Assigning cluster no to core and border point

- Eliminate noise points
- Perform clustering on the remaining points

$current\_cluster\_label \leftarrow 1$
for all core points **do**
  if the core point has no cluster label **then**
    $current\_cluster\_label \leftarrow current\_cluster\_label + 1$
    Label the current core point with cluster label $current\_cluster\_label$
  end if
  for all points in the Eps-neighborhood, except i^{th} the point itself **do**
    if the point does not have a cluster label **then**
      Label the point with cluster label $current\_cluster\_label$
    end if
  end for
end for

DBSCAN Algorithm

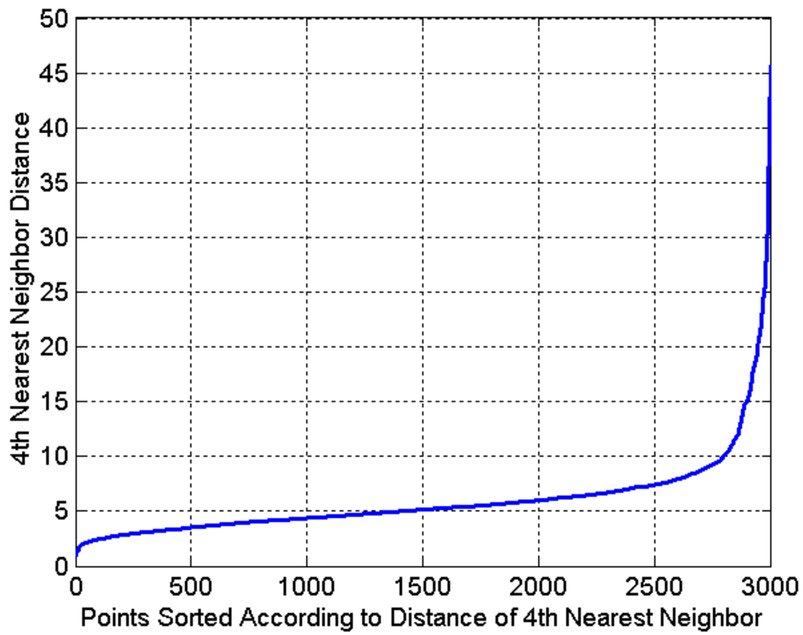
- **Time Complexity**
  - O(N x time to find points in Eps-neighbourhood)
  - where N is the no of points
  - Worst case $O(N^2)$
  - **KD-trees,** allow efficient retrieval of all points within given distance of a specified point in O(N logN)

- **Space Complexity**
  - O(N)

DBSCAN: Determining EPS and MinPts

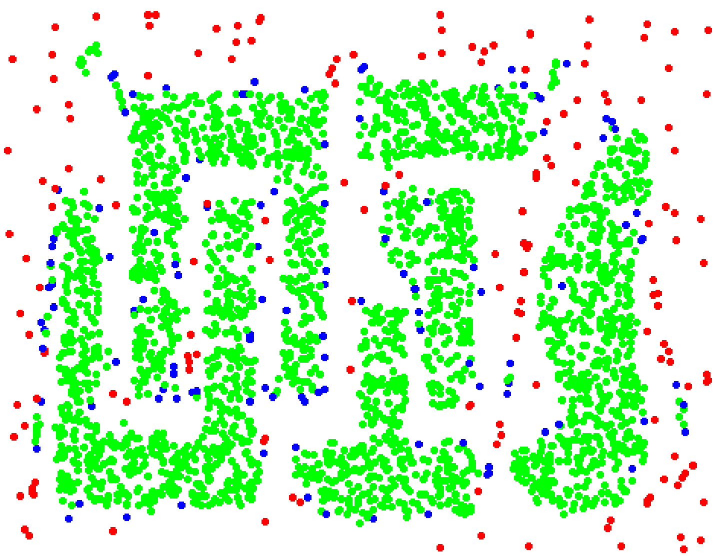
- **Idea:** For points in a cluster, their k^{th} nearest neighbors are at roughly the same distance
- **Noise points** have the k^{th} nearest neighbor at farther distance
- **Plot sorted distance** of every point to its k^{th} nearest neighbor
- We expect to see a **sharp change at the value of k-dist** that corresponds to a suitable value of Eps
- We can select this distance as Eps and k as minpts

Original db scan uses k=4 a reasonable no for points in 2-dimension



DBSCAN: Core, Border and Noise Points

Original Points

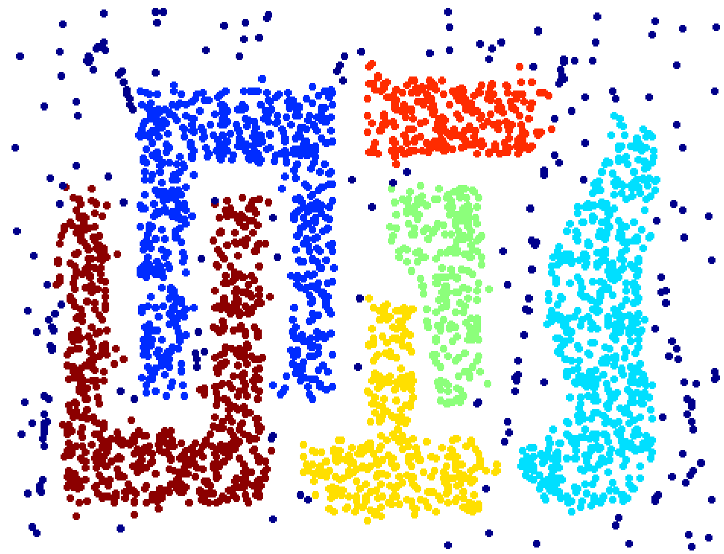


Point types: core, border and noise

Eps = 10, MinPts = 4

When DBSCAN Works Well

Original Points



Clusters

- Resistant to Noise
- Can handle clusters of different shapes and sizes

DBSCAN and Varying Densities

- ***Consider a dataset with high density regions A and B and Low density regions C and D***
  - If Eps threshold is low then
    - Dbscan can find C and D
    - But it will consider A , B and noise around it as one cluster
  - If Eps threshold is high then
    - Dbscan can detect find A and B as cluster and also noise around them
    - But it will mark C and D as noise too

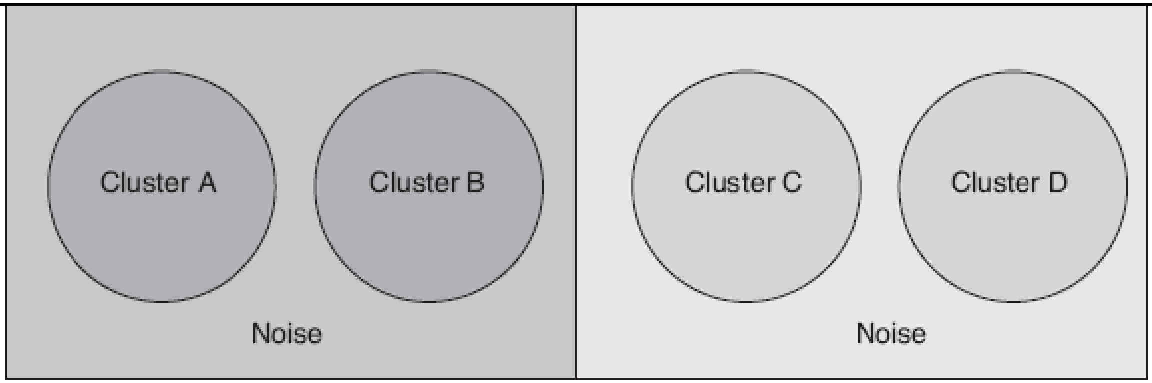
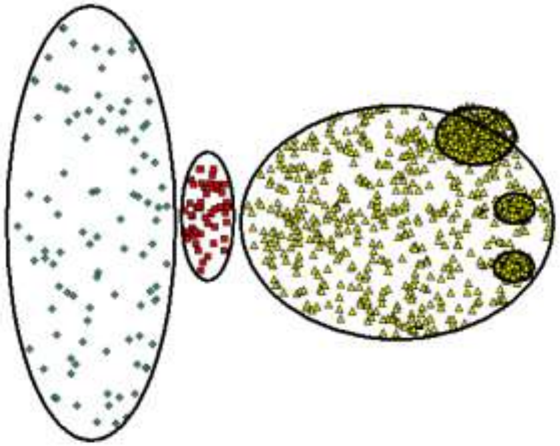


Figure 8.24. Four clusters embedded in noise.

DBSCAN Does NOT work Well

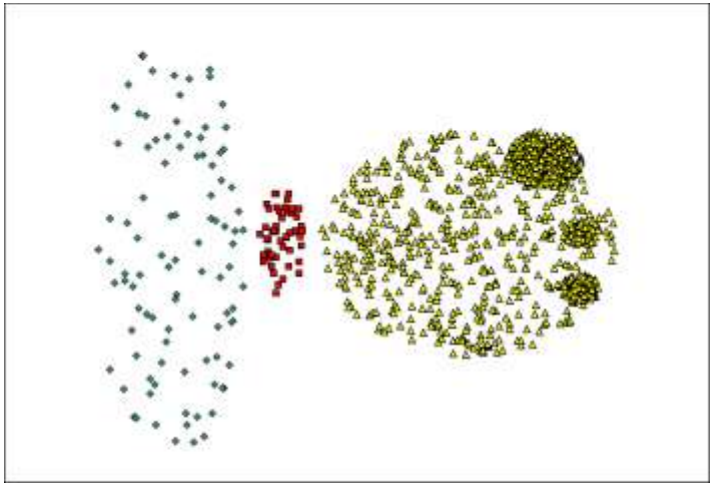
Can not handle varying densities.

When DBSCAN Does NOT Work Well

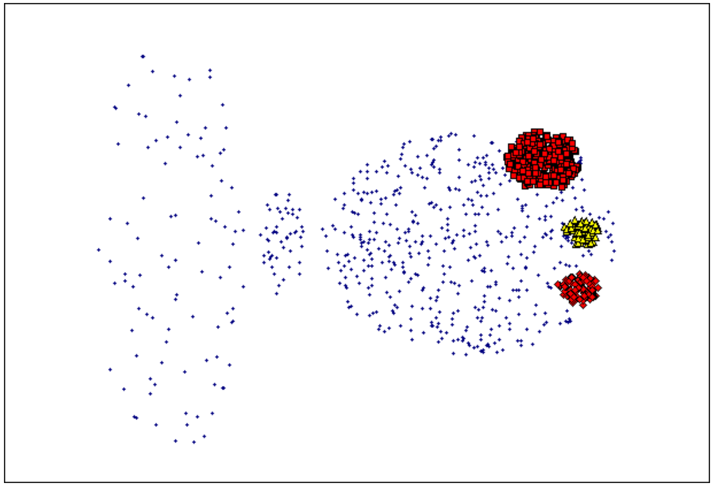


Original Points

- Varying densities
- High-dimensional data



(MinPts=4, Eps=9.75).



(MinPts=4, Eps=9.92)

DBSCAN: Sensitive to Parameters

Figure 8. DBScan results for DS1 with MinPts at 4 and Eps at (a) 0.5 and (b) 0.4.

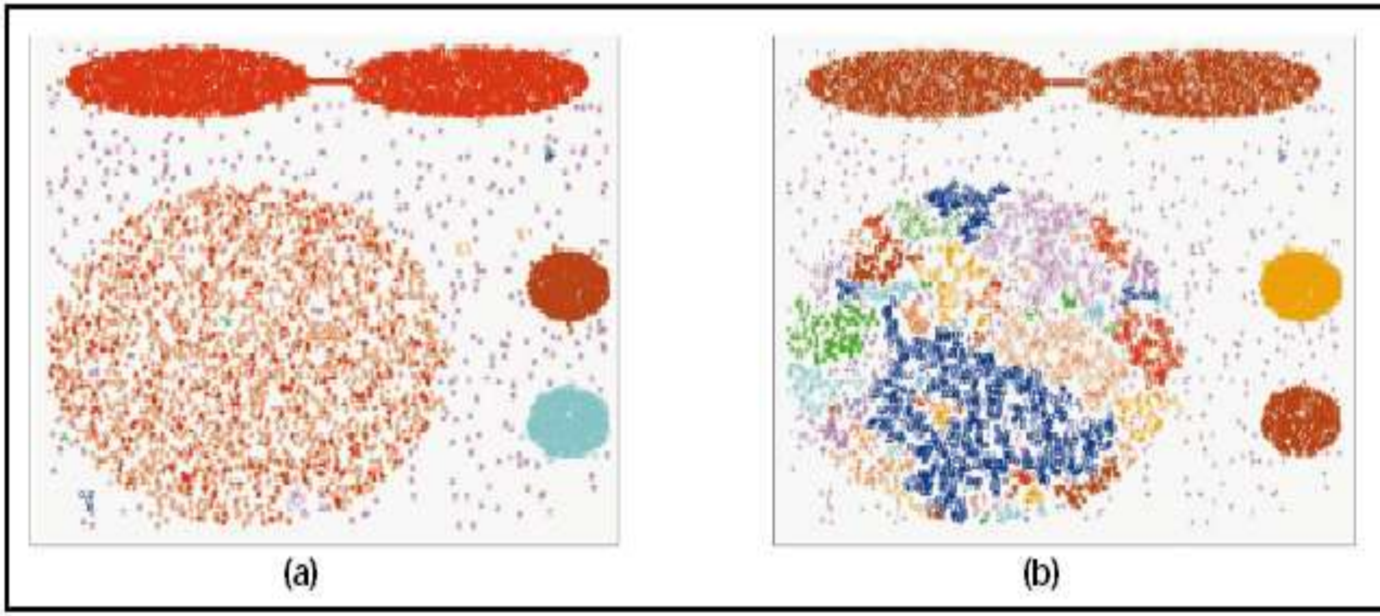
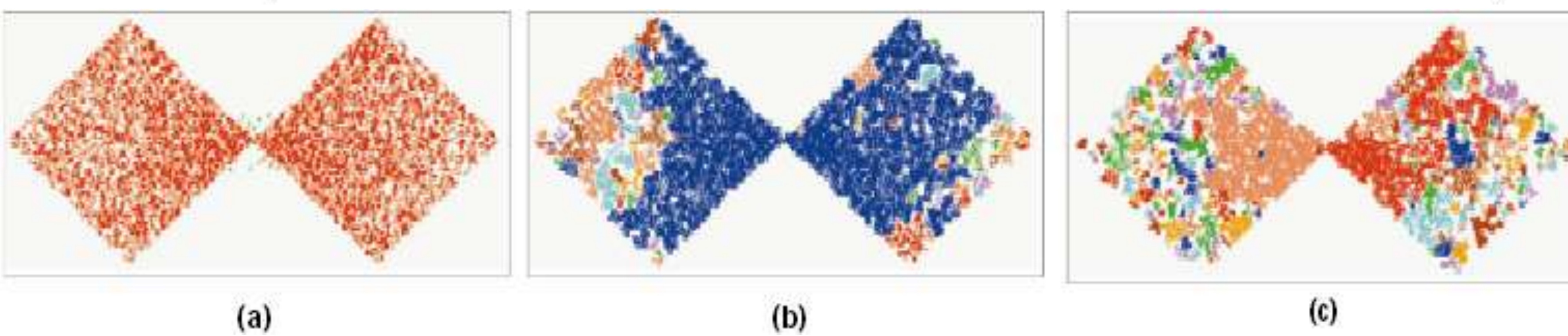


Figure 9. DBScan results for DS2 with MinPts at 4 and Eps at (a) 5.0, (b) 3.5, and (c) 3.0.



DBSCAN online Demo:

http://webdocs.cs.ualberta.ca/~yaling/Cluster/Applet/Code/Cluster.html

Problems and Challenges

- Considerable progress has been made in scalable clustering methods
  - Partitioning: k-means, k-medoids, CLARANS
  - Hierarchical: BIRCH, CURE
  - Density-based: DBSCAN, CLIQUE, OPTICS
  - Grid-based: STING, WaveCluster
  - **Model-based: Autoclass, Denclue, Cobweb**
- Current clustering techniques do not address all the requirements adequately
- Constraint-based clustering analysis: Constraints exist in data space (bridges and highways) or in user queries