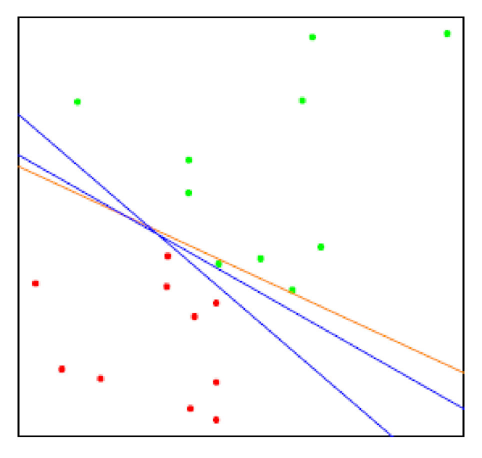


Parallel & Scalable Machine Learning

Introduction to Machine Learning Algorithms

Dr. –Ing. Gabriele Cavallaro

Postdoctoral Researcher
High Productivity Data Processing Group
Juelich Supercomputing Centre

Lecture 4 - 19/02/2020

UNSUPERVISED LEARNING – CLUSTERING

JÜLICH Forschungszentrum | JÜLICH SUPERCOMPUTING CENTRE

HELMHOLTZ AI | ARTIFICIAL INTELLIGENCE COOPERATION UNIT

PRACE PARTNERSHIP FOR ADVANCED COMPUTING IN EUROPE TRAINING PORTAL

UNIVERSITY OF ICELAND
SCHOOL OF ENGINEERING AND NATURAL SCIENCES
FACULTY OF INDUSTRIAL ENGINEERING,
MECHANICAL ENGINEERING AND COMPUTER SCIENCE

DEEP Projects

COURSE OUTLINE

- Parallel and Scalable Machine Learning Driven by HPC
- Introduction to Machine Learning Fundamentals
- Supervised Learning with a Simple Learning Model
- Artificial Neural Networks (ANNs)
- Introduction to Statistical Learning Theory
- Validation and Regularization
- Pattern Recognition Systems
- Parallel and Distributed Training of ANN
- Supervised Learning with Deep Learning
- Unsupervised Learning – Clustering
- Clustering with HPC
- Introduction to Deep Reinforcement Learning

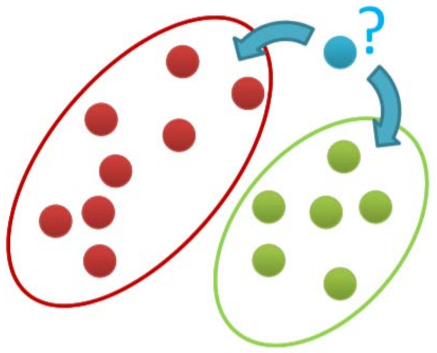
OUTLINE

- Unsupervised Learning
  - Unpromising approach?
- Clustering Approaches
- K-Means Algorithm
- DBSCAN Algorithm

MACHINE LEARNING

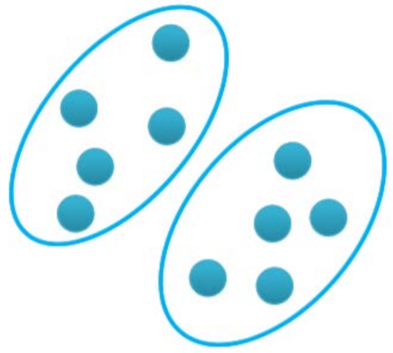
Form of Learning

- **Supervised learning:** correct responses for input data are given
  - "teacher" signal, correct "outcomes", "labels" for the data
  - Classic frameworks: **classification**, **regression**

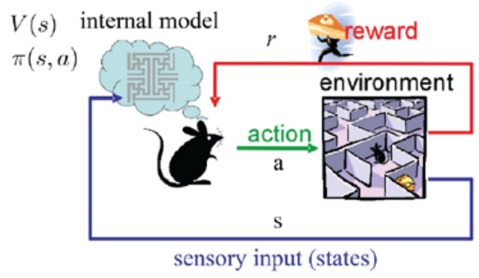


- **Unsupervised learning:** only data are given
  - Find "hidden" structure, patterns
  - Classical frameworks: **clustering**, **dimensionality reduction**

THIS LECTURE



- **Reinforcement learning:** data including (sparse) **reward** $\mathrm{r}(\boldsymbol{X})$
  - Discover actions $\boldsymbol{a}$ that minimize total future reward R
  - **Active** learning: experience depends on choice of $\boldsymbol{a}$



© MIT 6.S191: Introduction to Deep Learning
introtodeeplearning.com

USUPERVISED LEARNING

Unpromising approach?

- **Collecting** and **labeling** large set of sample patterns is **costly** and **challenging**
  - E.g., Earth observation data acquired from multi-source remote sensing instruments
  - Save your energy and time:
    - First, train a classifier on a small annotated dataset
    - Then “tuned up”: run the classifier without supervision on a large unlabeled set

[1] Fieldwork

- Reverse direction: first **train** with **large** amounts of **unlabeled data**
  - Then use **supervision** to label the **groupings found**
  - Appropriate for large “data mining” applications where the **contents** of large databases are **not known beforehand**

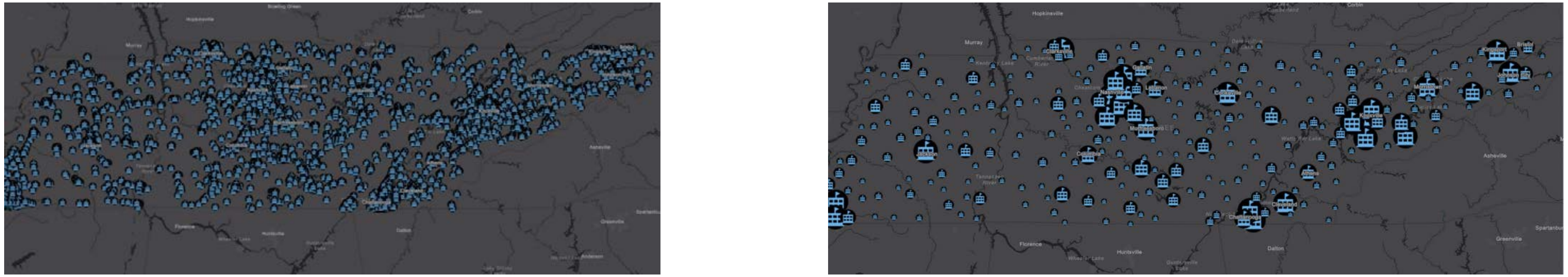
- In many applications, the **patterns** can **change with time**
  - Improve the performance by tracking these changes with a unsupervised classifier
  - E.g., Change detection of land cover classes

USUPERVISED LEARNING

Unpromising approach?

- Use unsupervised methods to find **features**
  - They can be useful for categorization

- For **early** stages of **investigations**
  - Gain some insight into the nature or structure of the data
  - Find distinct subclasses or similarities among patterns
  - This can drive the later design of the classifier



without clustering

with clustering

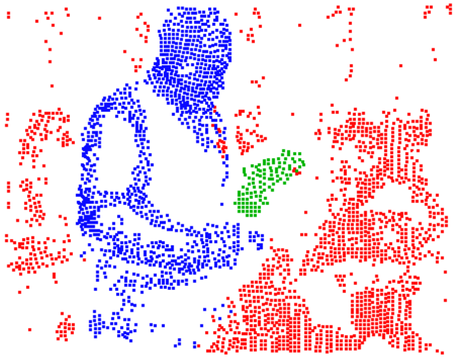
[2] Clustering Public Cooling Centers

USUPERVISED LEARNING

Problems

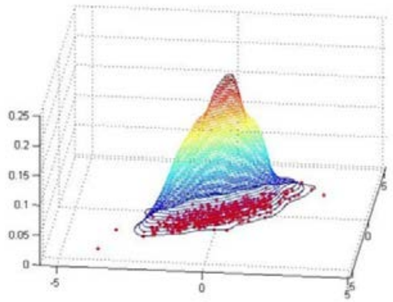
- **Clustering**: discover groups of similar examples within the data

THIS LECTURE



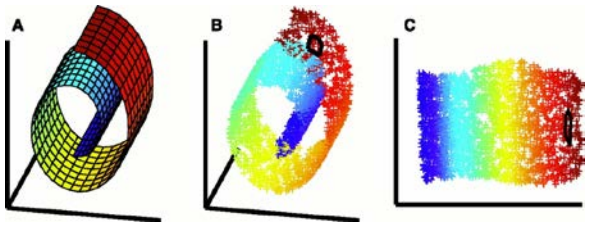
[3] T.Brox and J.Malik

- **Density estimation**: determine the distribution of data within the input space
  - Find a function that approximates the probability density of the data



[4] Machine learning & category recognition

- **Dimensionality reduction**: project the data from a high-dimensional space down to two or three dimensions
  - E.g., Visualization, condensation, compression

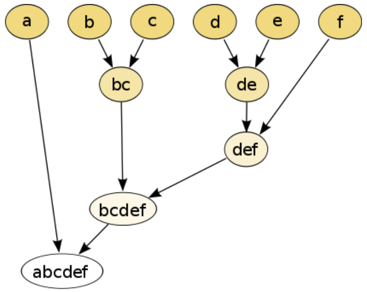


[5] Sam T. Roweis and Lawrence K. Saul

CLUSTERING

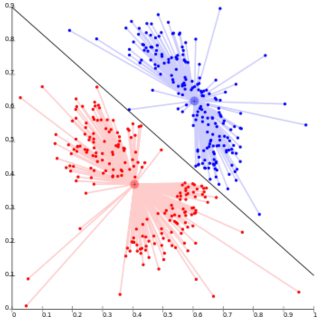
Approaches

- Clustering approaches can be categorized into four different approaches:



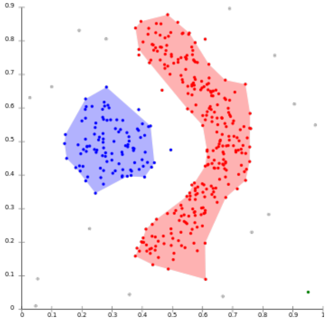
(hierarchical)

Clusters form a **hierarchy**. Can be computed bottom-up or top-down.



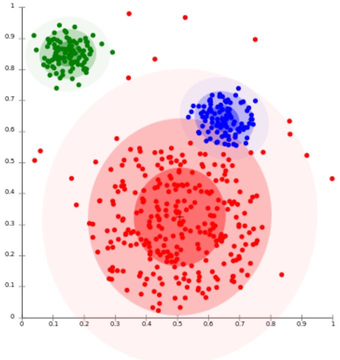
(centroid)

Similarity is derived by the **closeness** of a data point to the **centroid** of the clusters (**Iterative algorithms**).



(density)

Search the data space for areas of varied **density of data points**.



(distribution)

Based on the notion of how **probable** is it that all data points in the cluster belong to the **same distribution** (e.g., Normal, Gaussian).

- Terminology
  - **Flat clustering**: no inter-cluster structure
  - **Hard clustering**: items assigned to a unique cluster
  - **Soft clustering**: cluster membership is a real-valued function, distributed across several clusters

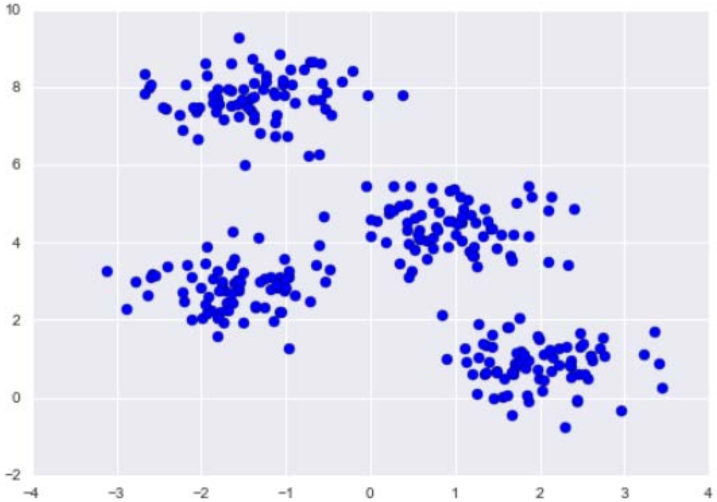
CLUSTERING

Problem Definition

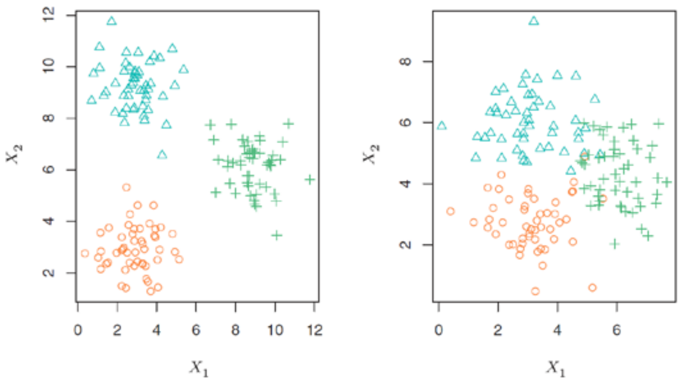
- Identify groups, or **clusters**, of data points in a multidimensional space
  - $\boldsymbol{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, \dots, \boldsymbol{x}_N\}$ be a set of N **training samples** (i.e., with no labels)
  - Observations of a random D-dimensional Euclidean variable $\boldsymbol{x}$

- Goal: **partition** the data set into some **number K of clusters**
  - Suppose that the value of K is **known**

- By eye, define clusters can be easy or ambiguous
  - **How** can one algorithm find **automatically** clusters?
  - The number of **possible combinations** of cluster assignments is **exponential** in the number of data points
    - An exhaustive search can be very expensive



[6] An Introduction to Statistical Learning



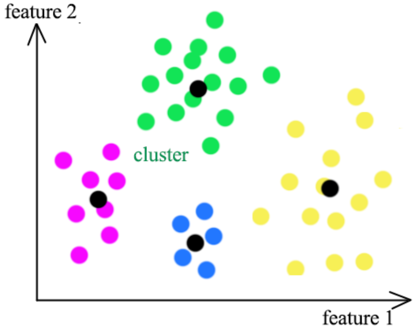
e.g., 2-dimensional datasets

CLUSTERING

Definition

- **Cluster**: group of data whose inter-point distances are small compared with the distances to points outside of the cluster

- **Centroids:** set of D-dimensional vectors $\boldsymbol{\mu}_k$ that represent that **centers of the clusters**
  - $\boldsymbol{\mu}_k$= prototype associated with the kth cluster (centroids can be artificial)
    - With $k = 1, \dots, K$



[7] Machine learning

- **Goal:** find an assignment of data points to clusters, as well as a set of vectors $\{\boldsymbol{\mu}_k\}$
  - Such that the **dissimilarity** (e.g., squared Euclidean distance) of each data point to its closest $\boldsymbol{\mu}_k$ is a **minimum**

CLUSTERING

1-of-*K* coding scheme - notation for the assignment of data points to clusters

- For each data point $\boldsymbol{x}_n$
  - Make a corresponding **set of binary indicator** variables $r_{nk} \in \{0,1\}$
    - $k = 1, \dots, K$ describe which of the K clusters the data point $\boldsymbol{x}_n$ is assigned to
  - If data point $\boldsymbol{x}_n$ is assigned to cluster k then $r_{nk} = 1$ and $r_{nj} = 0$ for $j \neq k$.

- Define an objective function (***distortion measure***)

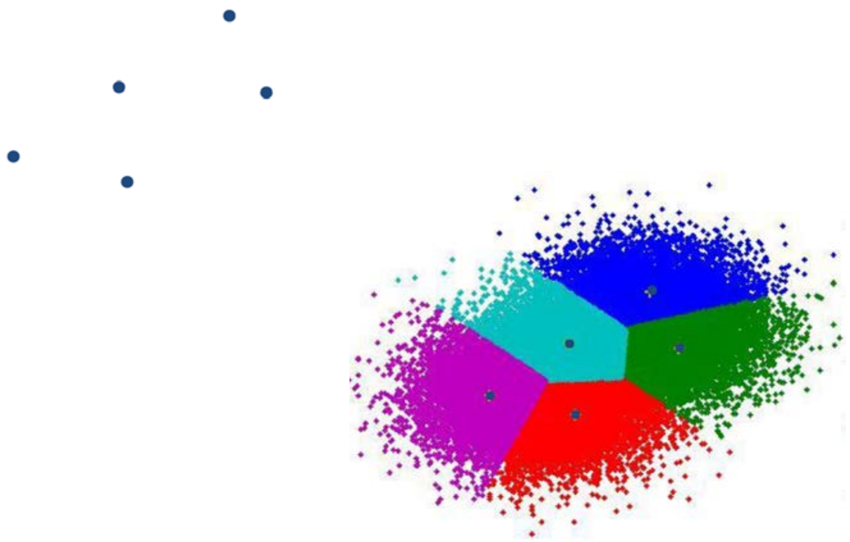
$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} || \boldsymbol{x}_n - \boldsymbol{\mu}_k| \Big|^2$$

  - Sum of the squares of the distances of each data point to its assigned vector $\boldsymbol{\mu}_k$

- Goal : minimize J by finding the best values of $\{r_{nk}\}$ and $\{\boldsymbol{\mu}_k\}$

K-MEANS CLUSTERING

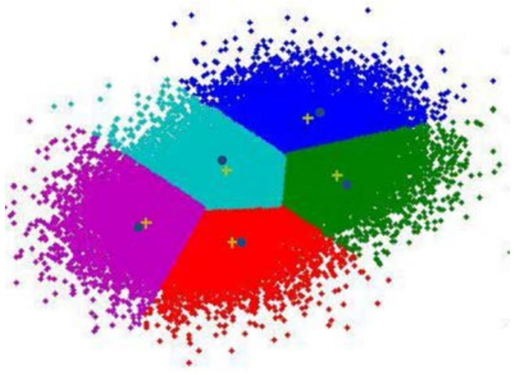
Procedure



- Set the **number** of **clusters** K and choose some initial values for $\boldsymbol{\mu}_k$
  - Picking the right number K is not trivial

- Assign in a random way a number from 1 to K to each observation

- **Iterate** a two-stage optimization until convergence (i.e., cluster assignments stop changing) - EM algorithm
  - Minimize J with respect to r_{nk} - E (expectation)
    - Keep $\boldsymbol{\mu}_k$ fixed and update r_{nk}
    - I.e., Assign each observation to the cluster k whose centroid $\boldsymbol{\mu}_k$ is closest
  - Minimize J with respect to $\boldsymbol{\mu}_k$ **-** M (maximization)
    - Keep r_{nk} fixed and update $\boldsymbol{\mu}_k$
    - I.e., For each of the K clusters, compute the cluster centroid $\boldsymbol{\mu}_k$



$$J = \sum_{n=1}^{N} \sum_{k=1}^{K} r_{nk} || \boldsymbol{x}_n - \boldsymbol{\mu}_k| |^2$$

K-MEANS CLUSTERING

Remarks

- The **two phases** of **re-assigning data points** to clusters and **re-computing the centroids** are repeated in turn **until** there is no further change in the **assignments**
  - Or until some maximum number of **iterations** is exceeded

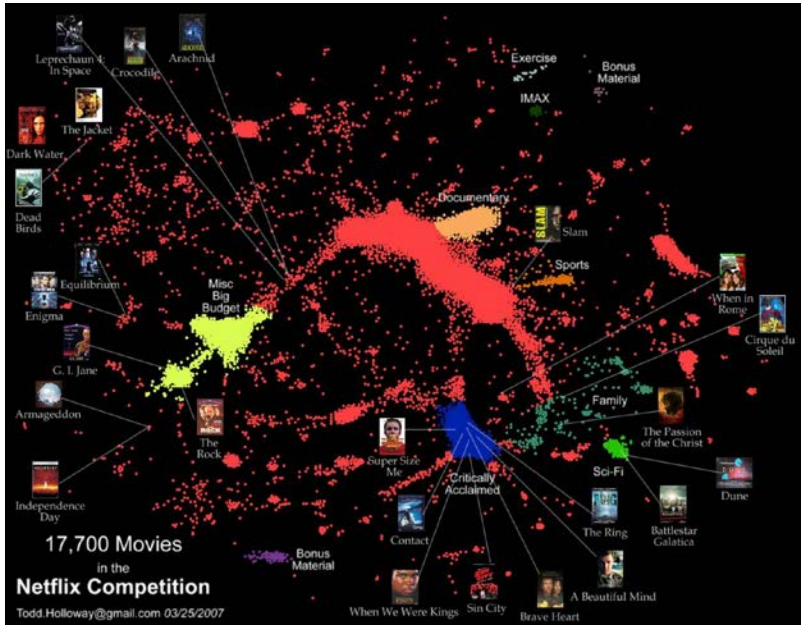
- Since each phase reduces the value of the objective function J
  - **Convergence** is assured
  - However, it may converge to a **local** rather than global minimum of J

- One notable feature: at each iteration, every data point is **assigned uniquely** to one of the clusters
  - I.e., Hard Clustering

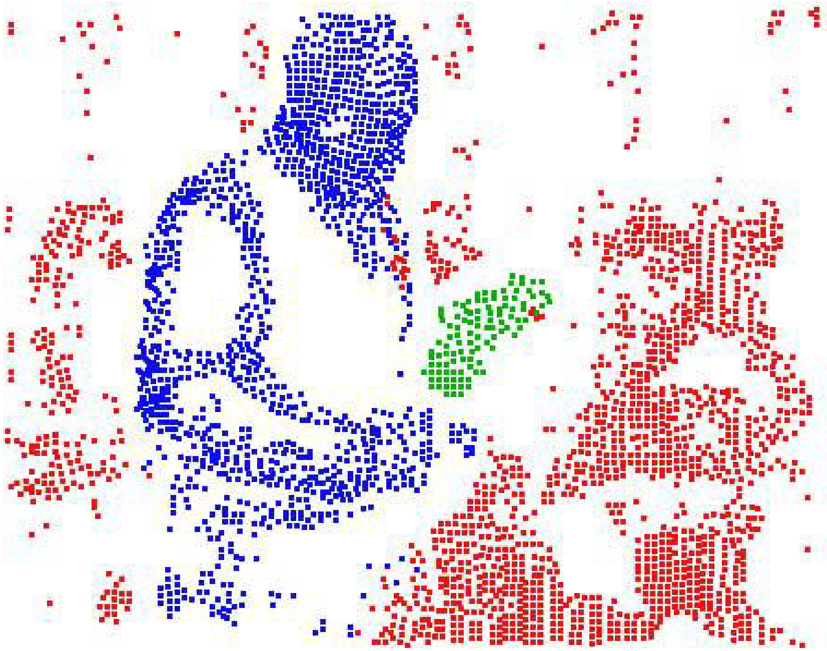
DBSCAN

Motivation

- Centroid-based clustering methods favor clusters that are **spherical**
  - They have great difficulty with anything else
- Bu in **real data** we have:



[8] Todd Holloway

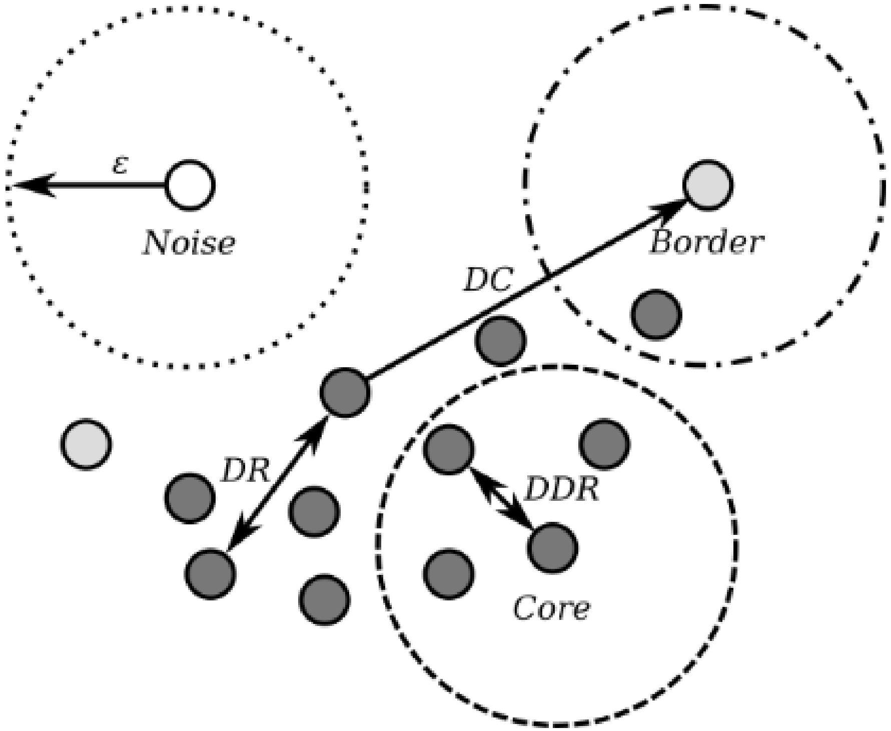


[3] T.Brox and J.Malik

DBSCAN

Density-based clustering

- Introduced 1996 and most cited clustering algorithm
- It follows the shape of **dense neighborhoods** of points
- Distinct Features
  - Clusters a variable number of clusters
  - Forms **arbitrarily shaped clusters** (except ‘bow ties‘)
  - **Identifies** inherently also **outliers/noise**
- Understanding Parameters
  - Looks for a similar points within a given search radius
    - Parameter **epsilon (*ε*)**
  - A cluster consist of a given minimum number of points
    - Parameter **minPoints**



(MinPoints = 4)

(DR = Density Reachable)

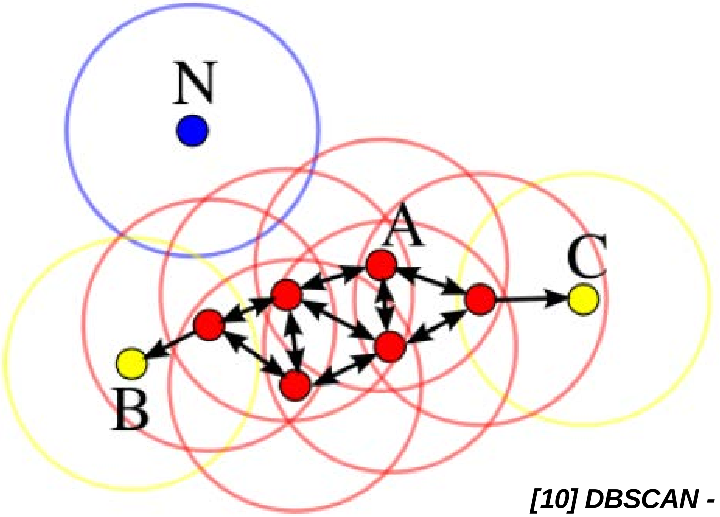
(DDR = Directly Density Reachable)

(DC = Density Connected)

[9] Ester et al.

DBSCAN

- **Core points** can **directly reach** neighbors in their **epsilon (*ε*)**-sphere
  - Non-core points cannot directly reach another point
- A point q is **Density Reachable (DR)** from p
  - If there is a series of points $p = p_1, p_2, \ldots, p_n$ such that p_{i+1} is directly reachable from p_i
  - If a point is DR from a cluster-point, it is part of the cluster as well
- **All points not DR** from any other points are **outliers**



(MinPoints = 4)

[10] DBSCAN - Wikipedia

Points A are density connected (core points)

Points B, C are density reachable

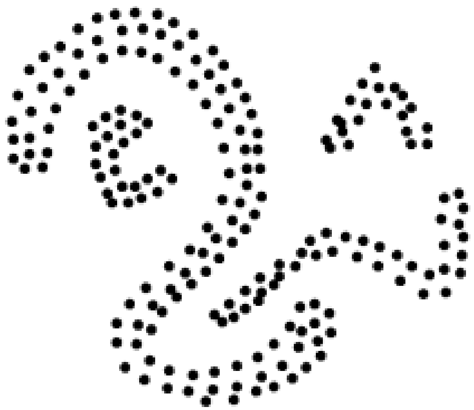
A, B, C for a single cluster

N is considered as noise

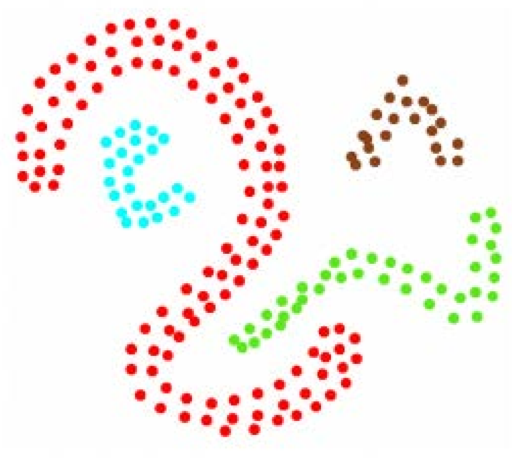
DBSCAN ALGORITHM

Non-Trivial Example

- Compare K-Means vs. DBSCAN – How would K-Means work?



Unclustered
Data

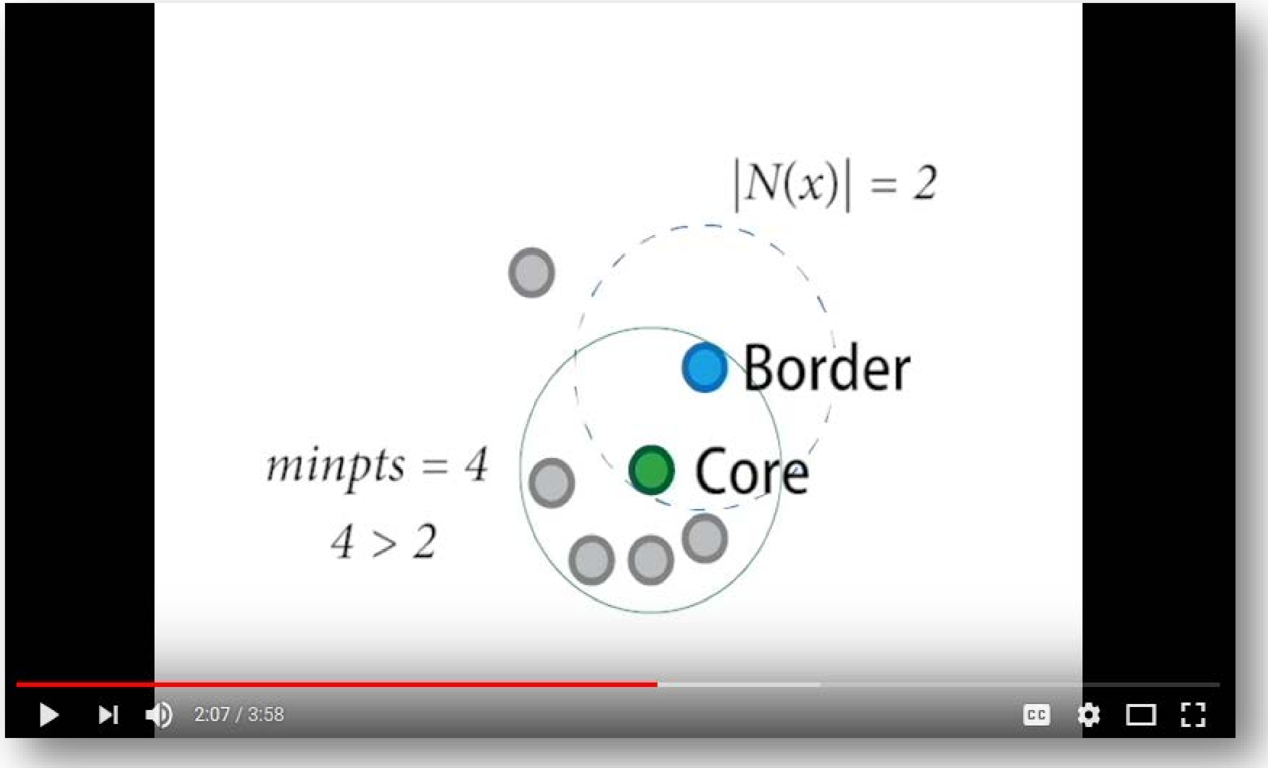


Clustered
Data

- **DBSCAN forms arbitrarily shaped clusters (except ‘bow ties‘) where other clustering algorithms fail**

DBSCAN

Clustering



[4] DBSCAN, YouTube Video

REFERENCES

- [1] Geography Fieldwork
  Online: https://www.field-studies-council.org/geography-fieldwork/
- [2] Clustering in ArcGIS
  Online: https://www.esri.com/arcgis-blog/products/mapping/mapping/clustering-in-arcgis-online-enables-data-exploration-september-2017/
- [3] T.Brox and J.Malik, ‘’Object segmentation by long term analysis of point trajectories’’, European Conference on Computer Vision (ECCV), Springer, 2010.
- [4] Machine learning & category recognition, Cordelia Schmid and Jakob Verbeek
  Online: https://slideplayer.com/slide/7085756/
- [5] Sam T. Roweis and Lawrence K. Saul, ‘’ Nonlinear Dimensionality Reduction by Locally Linear Embedding’’, in Science, vol. 290, no. 5500, pp. 2323-2326, 2000.
- [6] An Introduction to Statistical Learning with Applications in R
  Online: http://www-bcf.usc.edu/~gareth/ISL/index.html
- [7] Machine learning - Clustering, Density based clustering and SOM
  Online: https://jhui.github.io/2017/01/15/Machine-learning-clustering/
- [8] Todd Holloway, ‘’Ensemble Learning’’
  Online: https://www.slideshare.net/butest/ensemble-learning-featuring-the-netflix-prize-competition-and
- [9] Ester, Martin, et al. "A density-based algorithm for discovering clusters in large spatial databases with noise." Kdd. Vol. 96. 1996.
- [10] DBSCAN - Wikipedia
  Online: https://en.wikipedia.org/wiki/DBSCAN

REFERENCES

- [11] YouTube Video, ‘CSCE 420 Communication Project – DBSCAN‘, Online: https://www.youtube.com/watch?v=5E097ZLE9Sg