Analytics and Visualization of Big Data

Fadel M. Megahed

Lecture 15: Clustering (Discussion Class)

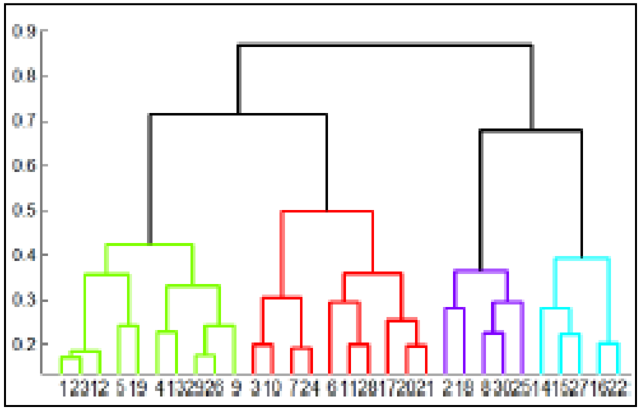
AUBURN UNIVERSITY

SAMUEL GINN
COLLEGE OF ENGINEERING

Department of Industrial and Systems Engineering

Spring 13

Refresher: Hierarchical Clustering

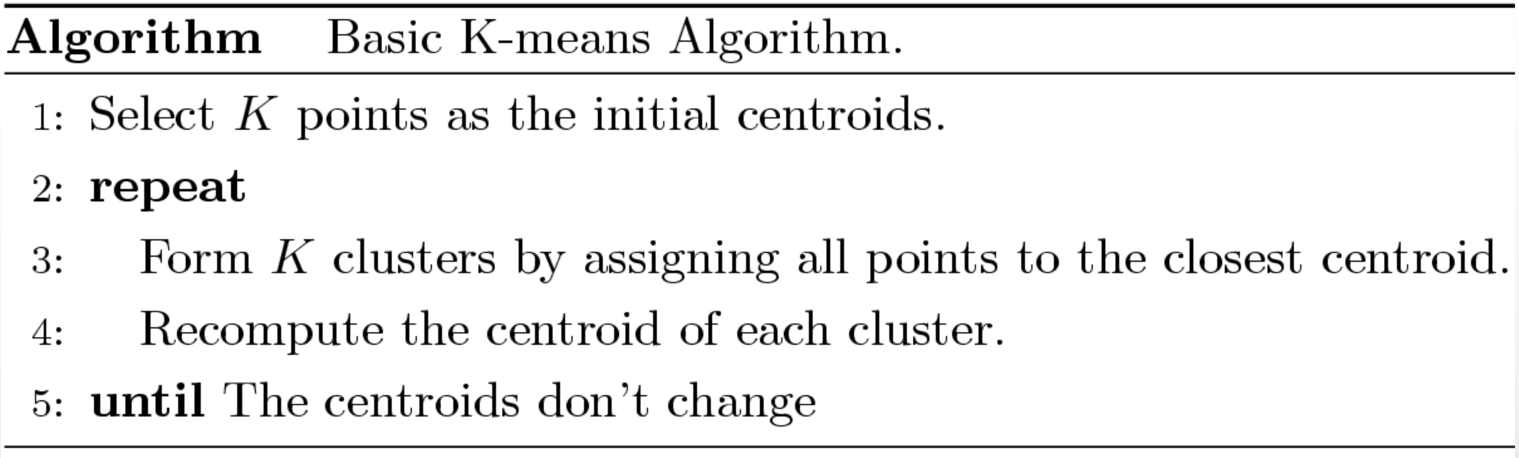


- **Key Operation:**
  - Repeatedly combine the two nearest clusters into one (for bottom up)

- **Important questions:**
  1. How will clusters be represented? → (Euclidean and Non-Euclidean Distances)
  2. How will we choose which two clusters to merge? → Measure cluster dist. by dist. of centroids
  3. When will we stop combining clusters? → When combining → inadequate cluster

Refresher: K-Means Clustering

- Most widely used clustering algorithm. It follows a very simple procedure whose main characteristics are:
  - Assumes Euclidean space/distance
  - Start by picking *k*, the number of clusters
  - Initialize clusters by picking one point per cluster
    - Example: Pick one point at random, then **k-1** other points, each as far away as possible from the previous points



Algorithm Basic K-means Algorithm.

```
Select $K$ points as the initial centroids.
repeat
    Form $K$ clusters by assigning all points to the closest centroid.
    Recompute the centroid of each cluster.
until The centroids don't change
```

Topics for DISCUSSION in Today’s Class

- How do you cluster when the memory does not fit all the data points?
- How do you cluster such that you are able to detect non-convex clusters?

Based on Tuesday’s class, you have all expressed that clustering is so intuitive (which is true), the objective behind this experiment is to think about the basic thought process for some of the more complex clustering algorithms

Overview of Topics Covered in Chapter 7

Hierarchical Clustering

- In Euclidean Space
- Efficiency
- In Non-Euclidean Spaces

K-Means

- Basics
- Initialization
- Picking the Right Value of K
- BFR Algorithm

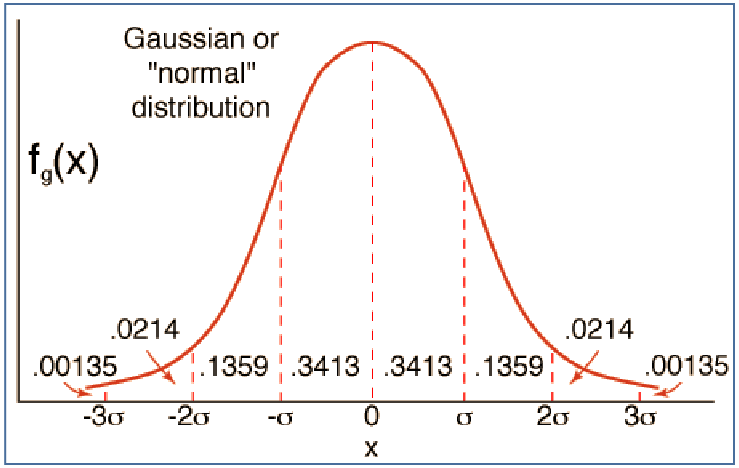
The Cure Algorithm

- Initialization
- Completion of the CURE Algorithm

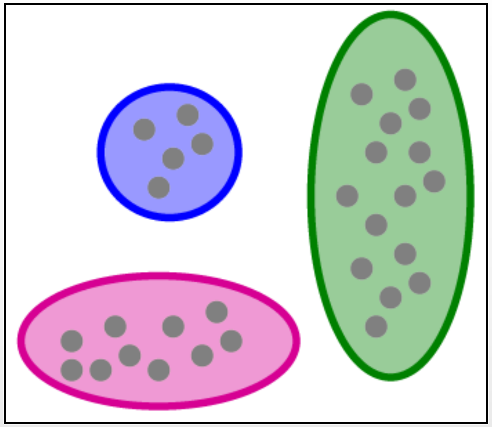
BFR Algorithm*

- **BFR** [Bradley-Fayyad-Reina] is a variant of *k*-means designed for very large (disk-resident) datasets

- **Assumes** that clusters are normally distributed around a centroid in a Euclidean space
  - Standard deviations in different dimensions may vary
    - Clusters are axis-aligned ellipses
  - For every point we can quantify the **likelihood** that it belongs to a particular cluster



Source: http://hyperphysics.phy-astr.gsu.edu/%E2%80%8Chbase/math/gaufcn.html



*Slides Adapted from Jure Leskovic, Stanford CS246, Lecture Notes, see http://cs246.stanford.edu

BFR Algorithm*: The Process

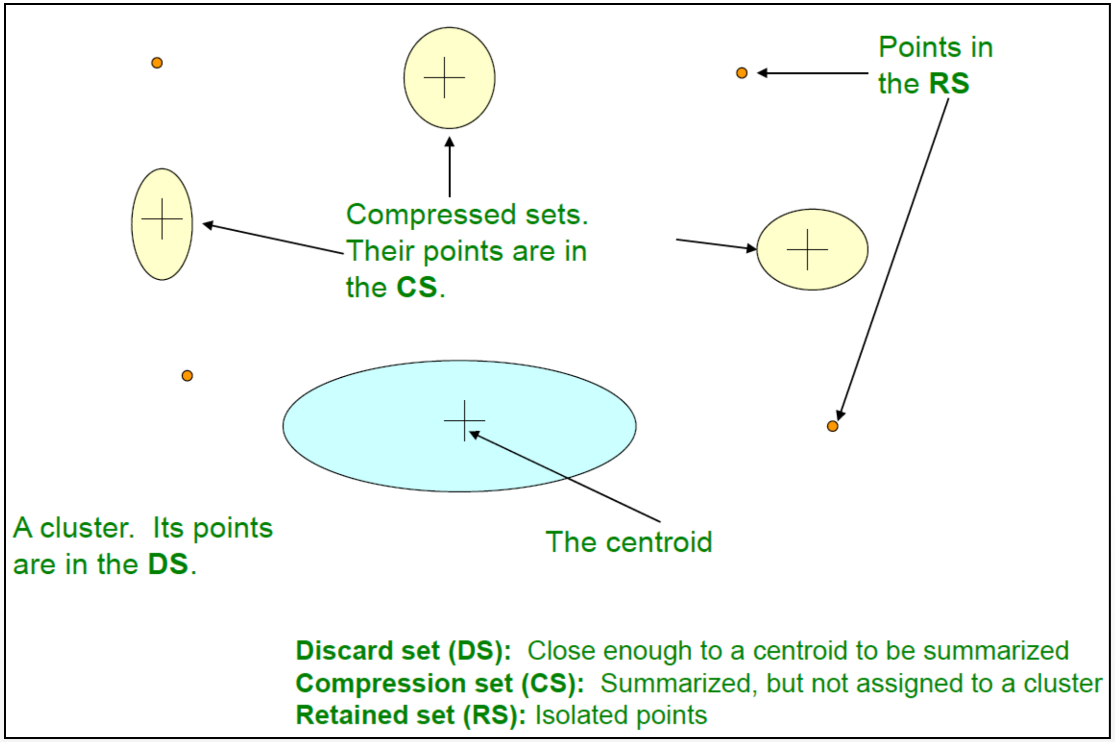
- **Points are read one main-memory-full at a time**

- Most points from previous memory loads are summarized by **simple statistics**

- To begin, from the initial load we select the initial ***k*** centroids by some sensible approach, e.g.:
  - Take ***k*** random points
  - Take a sample; pick a random point, and then ***k*–*1*** more points, each as far from the previously selected points as possible (works better than taking the ***k*** random points)

BFR Algorithm*: Three Classes of Points

- 3 sets of points which we keep track of:
  - **Discard set (DS):**
    - Points close enough to a centroid to be summarized
  - **Compression set (CS):**
    - Groups of points that are close together but not close to any existing centroid
    - These points are summarized, but not assigned to a cluster
  - **Retained set (RS):**
    - Isolated points waiting to be assigned to a compression set



Points in the **RS**

Compressed sets. Their points are in the **CS**.

A cluster. Its points are in the **DS**.

The centroid

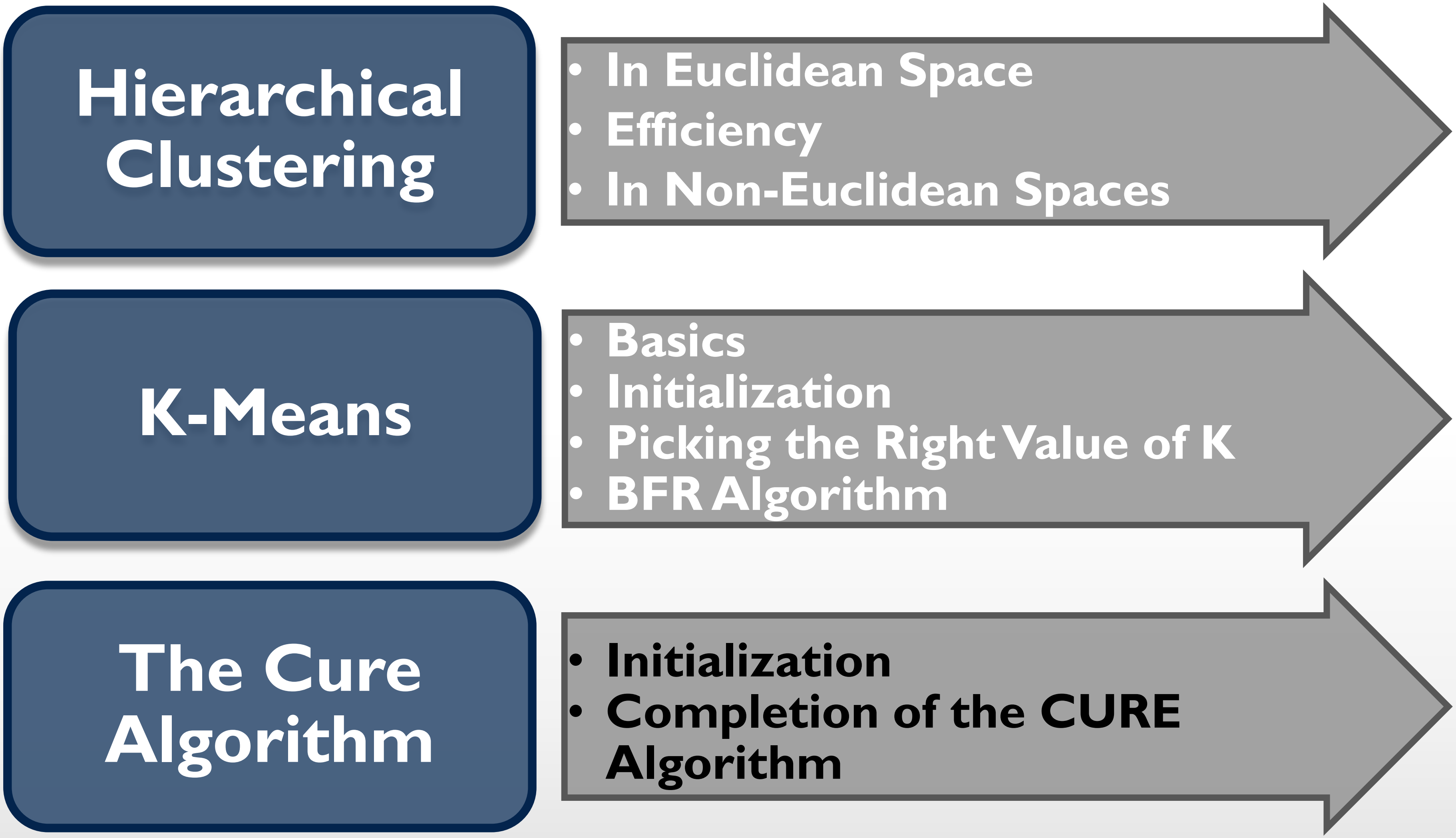
Discard set (DS): Close enough to a centroid to be summarized
Compression set (CS): Summarized, but not assigned to a cluster
Retained set (RS): Isolated points

Topics for DISCUSSION in Today's Class

- How do you cluster when the memory does not fit all the data points?
- How do you cluster such that you are able to detect non-convex clusters?

Based on Tuesday's class, you have all expressed that clustering is so intuitive (which is true), the objective behind this experiment is to think about the basic thought process for some of the more complex clustering algorithms

Overview of Topics Covered in Chapter 7



Hierarchical Clustering

- In Euclidean Space
- Efficiency
- In Non-Euclidean Spaces

K-Means

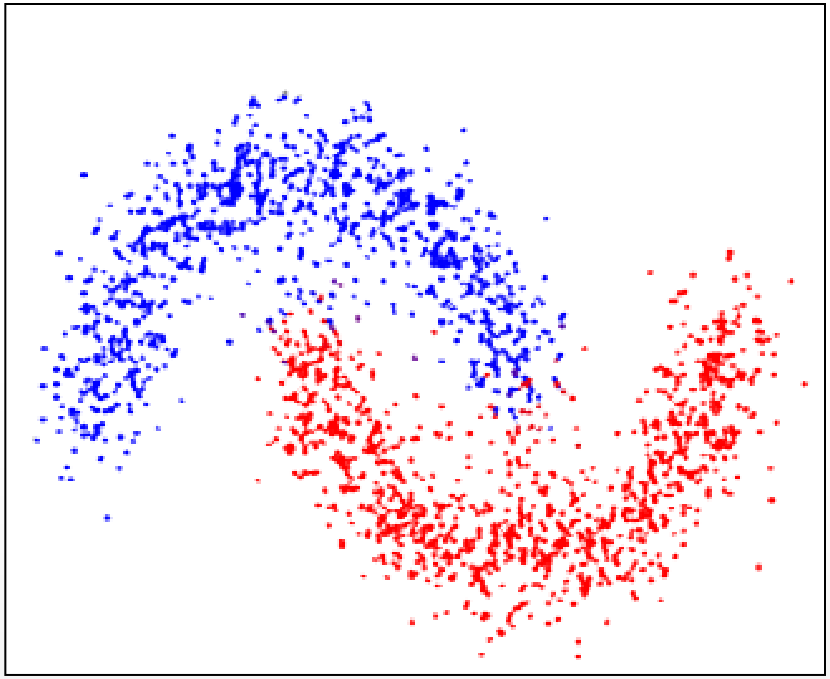
- Basics
- Initialization
- Picking the Right Value of K
- BFR Algorithm

The Cure Algorithm

- Initialization
- Completion of the CURE Algorithm

The CURE Algorithm

- **Problem with BFR/*k*-means:**
  - Assumes clusters are normally distributed in each dimension
  - And axes are fixed – ellipses at an angle are ***not OK***



- **CURE (Clustering Using REpresentatives):**
  - Assumes a Euclidean distance
  - Allows clusters to assume any shape
  - **Uses a collection of representative points to represent clusters**

CURE Algorithm

Pass 1:

Pick a random sample of points that fit in memory

1. **Initial clusters:**
    - Cluster these points hierarchically – group nearest points/clusters
2. **Pick representative points:**
    - For each cluster, pick a sample of points, as dispersed as possible
    - From the sample, pick representatives by moving them (say) 20% toward the centroid of the cluster

Pass 2:

1. **Now, rescan the whole dataset and visit each point *p* in the data set**
2. **Place it in the "closest cluster"**
    - Normal definition of "closest": that cluster with the closest (to *p*) among all the representative points of all the clusters

1] Randomly pick 100,000 points & form clusters using one of the approaches we talked about (hier..)

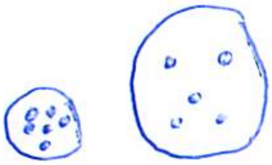
→ # clusters (n_i)

2] Represent the clusters w/ their centroids

3] Repeat for a set # iterations (i=100)
→ 100 sets of ~~c~~centroids

→

4] Avg/Cluster your 100 sets of centroids



how to avg/cluster centroids?

1] Min. Distance

2] Normalize dimensions

Area for Improvment:

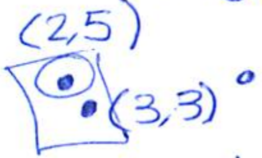
1) variability within a cluster

2) Phil's: - Randomly select & remove pool

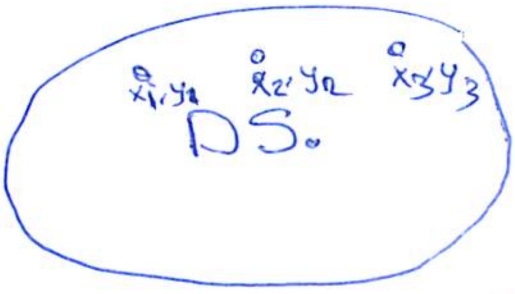
3) Make a dist. assmp. (Jessica's Method)

4) Take a sample & figure out a dist.

- how do you ~~need to~~ represent cluster ?
  - → Just keeping centroid



- Variability cluster
  - N is in a cluster
  - Variability of the cluster



3 things:

→ N

→ SUM vector

→ SUMSQ vector

centroid   SUM $SUM = \frac{x_1 + x_2 + \cdots + x_N}{N}$

SUMSQ ⇒ variance of the cluster

↳ $Var(Cluster_i) = \left[\frac{SUMSQ_i}{N}\right] - \left[\frac{SUM_i}{N}\right]^2$

Convex set?