Clustering data at scale

Dan Filimon, Politehnica University Bucharest
Ted Dunning, MapR Technologies

whoami

- Soon-to-be graduate of Politehnica Univerity of Bucharest
- Apache Mahout committer
- This work is my senior project
- Contact me at
  - dfilimon@apache.org
  - dangeorge.filimon@gmail.com

Agenda

- Data
- Clustering
- k-means
- Improvements
- Large scale
- k-means as a map-reduce
- streaming k-means
- MapReduce & Storm
- Results

Full version at http://goo.gl/n3n8S

Data

- Real-valued vectors (or anything that can be encoded as such)
- Think of rows in a database table
- Can be: documents, web pages, images, videos, users, DNA

The problem

Group **n** **d**-dimensional datapoints into **k** disjoint sets to minimize

$$\sum_{\mathbf{c_1}}^{\mathbf{c_k}} \left(\sum_{\mathbf{x_{ij}} \in X_i} dist(\mathbf{x_{ij}}, \mathbf{c_i}) \right)$$

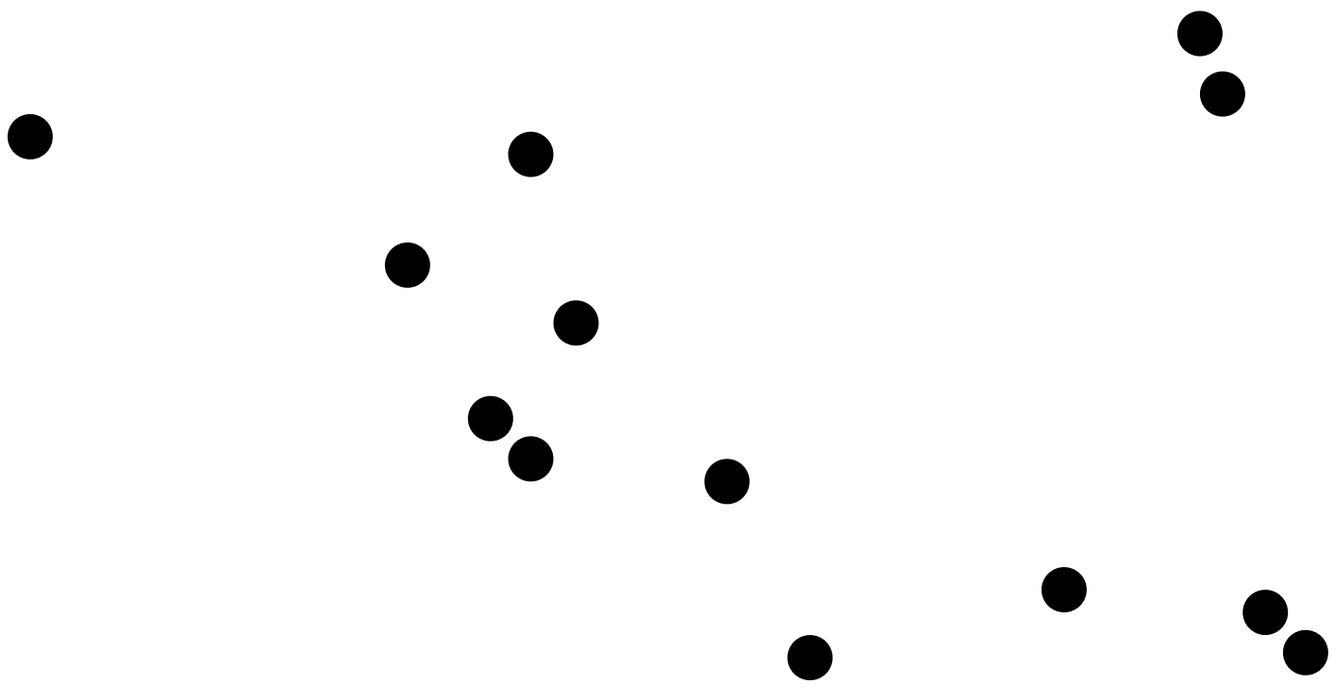
X_i is the i^{th} cluster

$\mathbf{c_i}$ is the centroid of the i^{th} cluster

$\mathbf{x_{ij}}$ is the j^{th} point from the i^{th} cluster

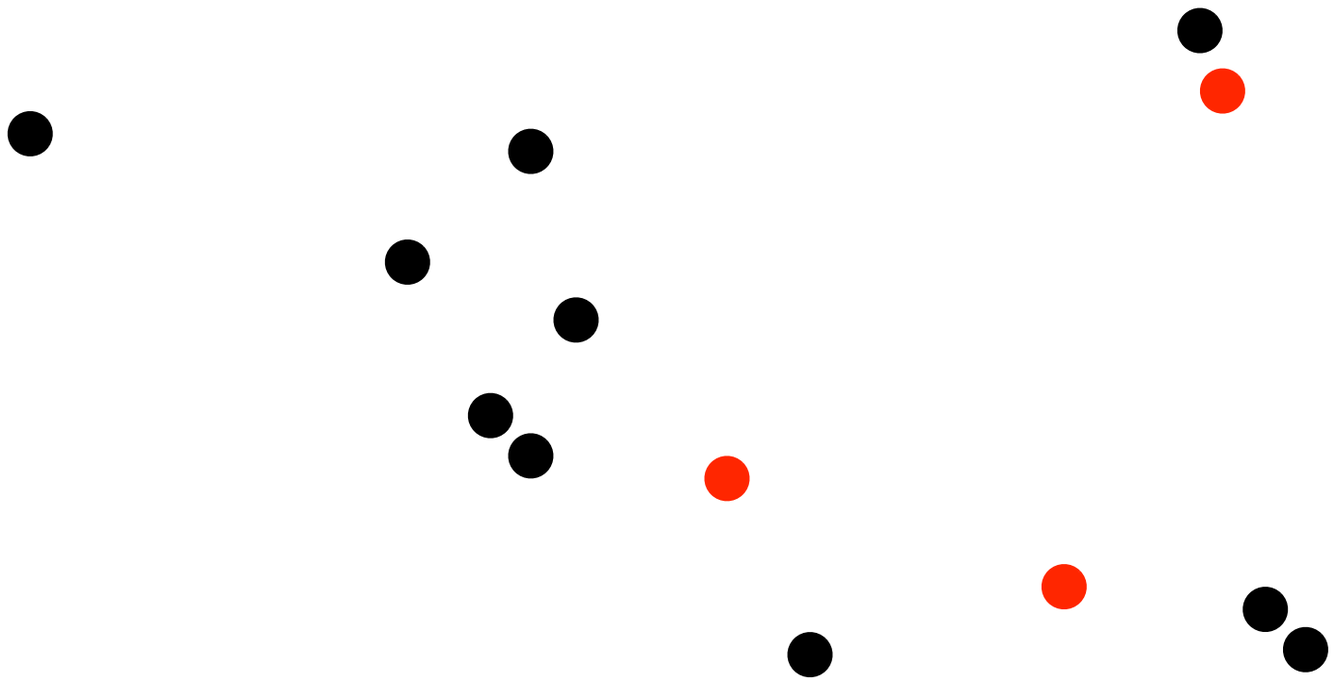
$$dist(\mathbf{x}, \mathbf{y}) = ||\mathbf{x} - \mathbf{y}||_2$$

k-means



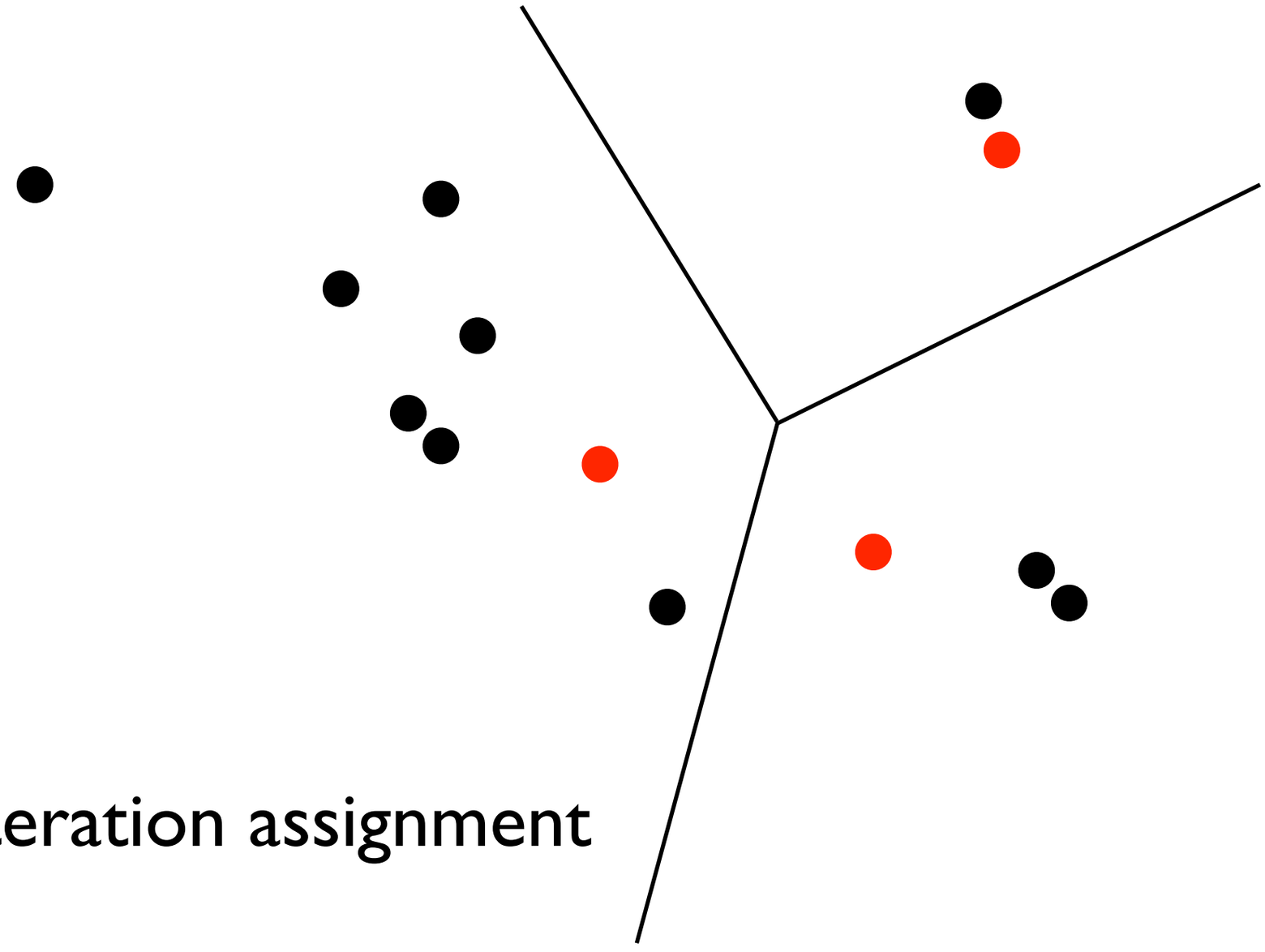
points to cluster

k-means



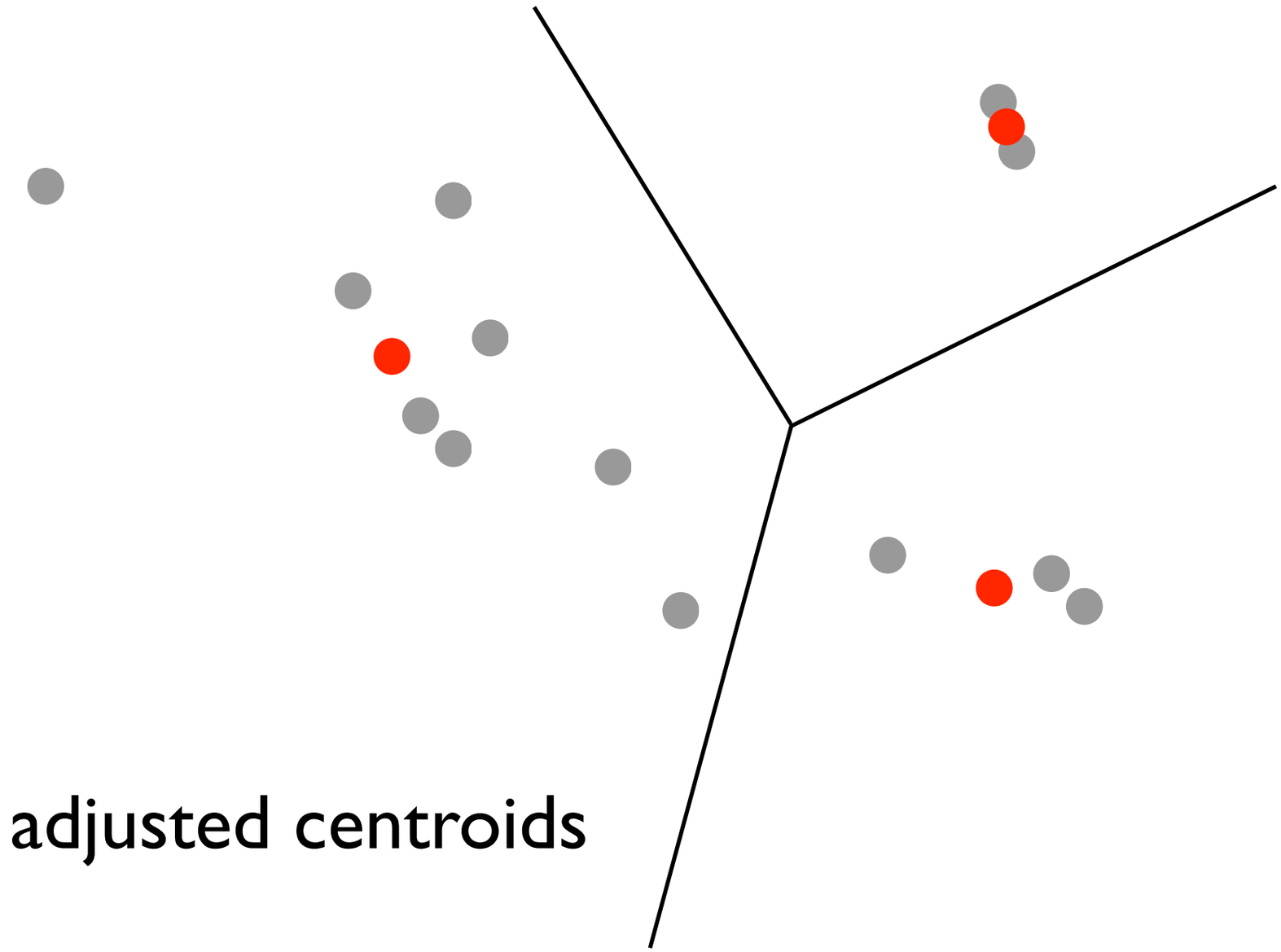
initial centroids

k-means



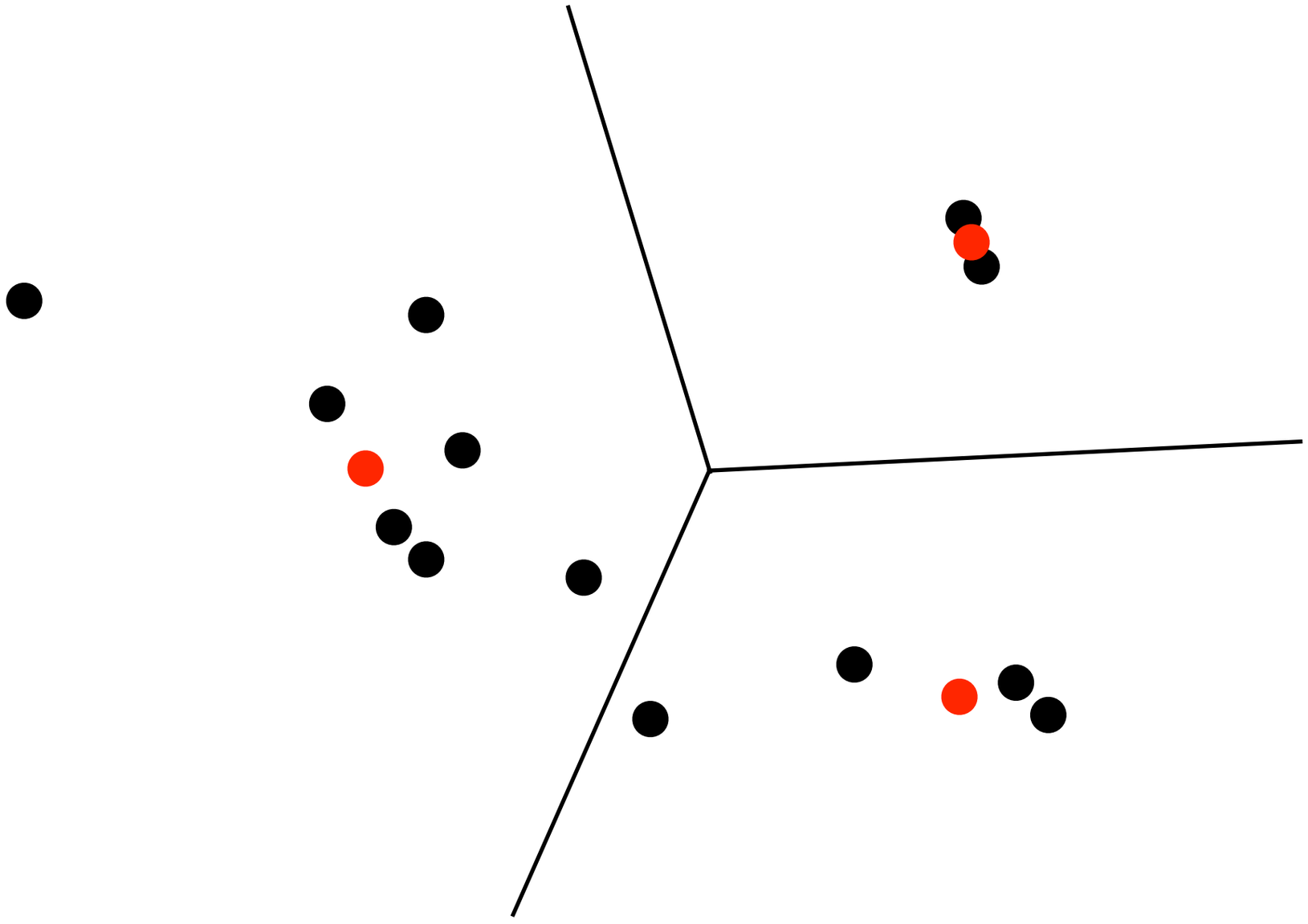
first iteration assignment

k-means



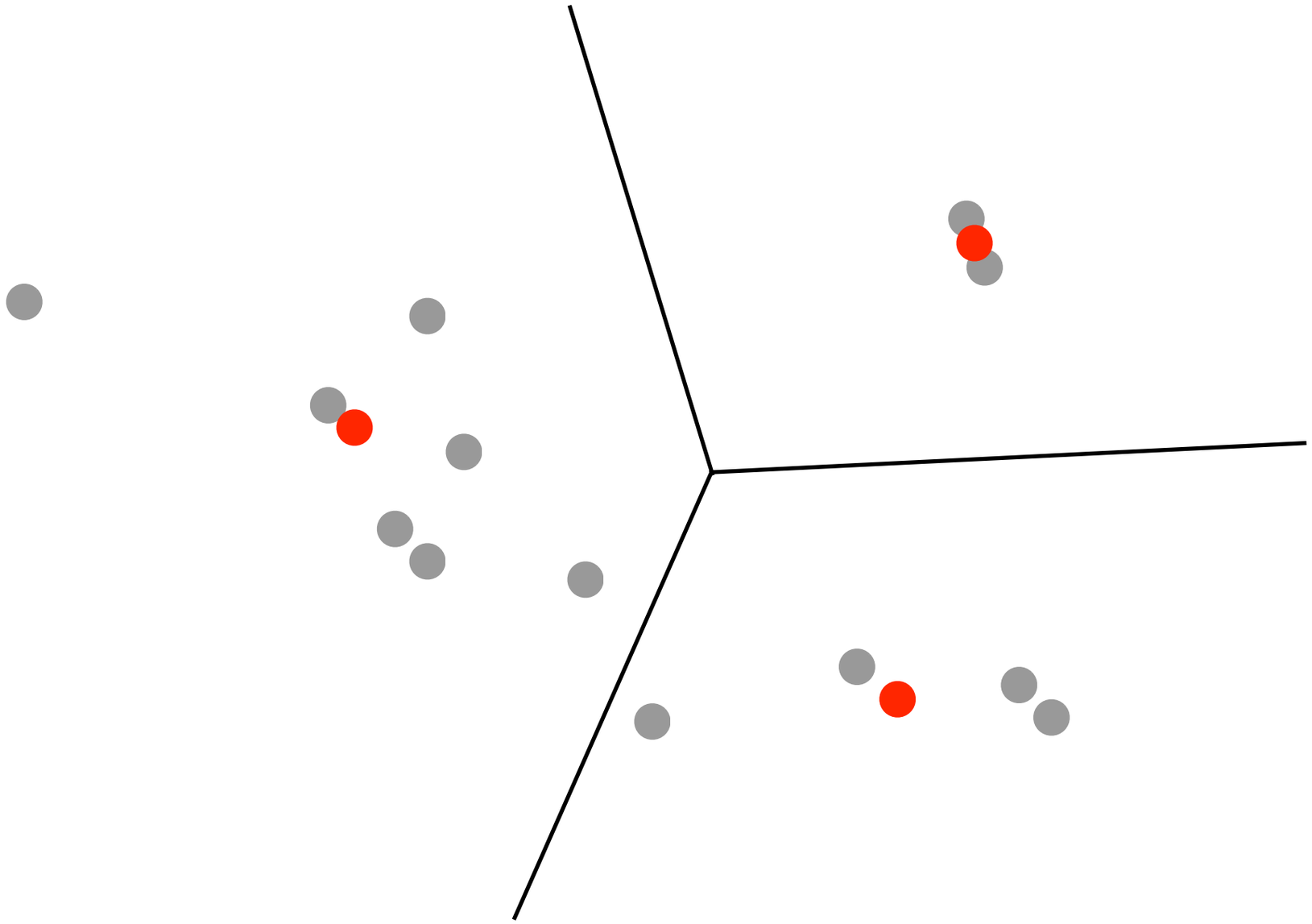
adjusted centroids

k-means



second iteration assignment

k-means



adjusted centroids

Details

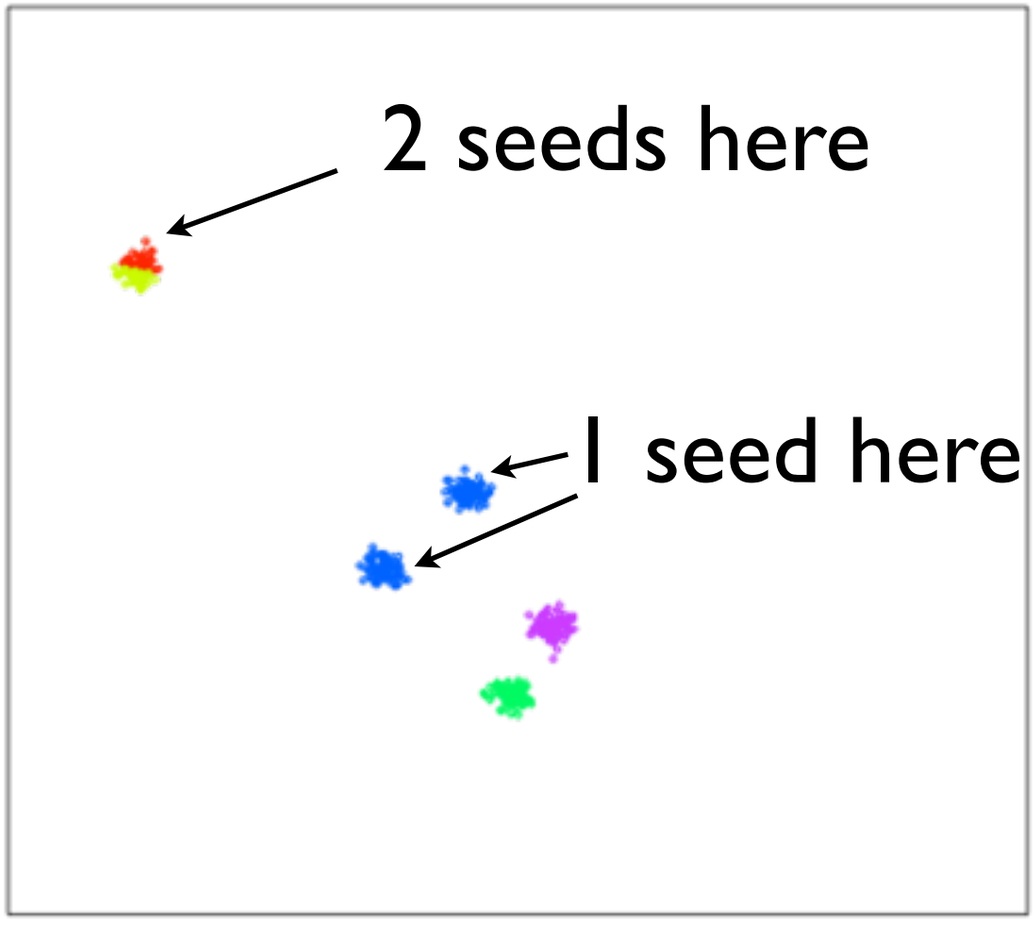
- Quality
  - How to initialize the centroids
  - When to stop iterating
  - How to deal with outliers
- Speed
  - Complexity of cluster assignment

Quality?

- Clustering is in the eye of the beholder
  - Total clustering cost and:
    - compact
    - well-separated
  - Dunn Index, Davies-Bouldin Index, etc.

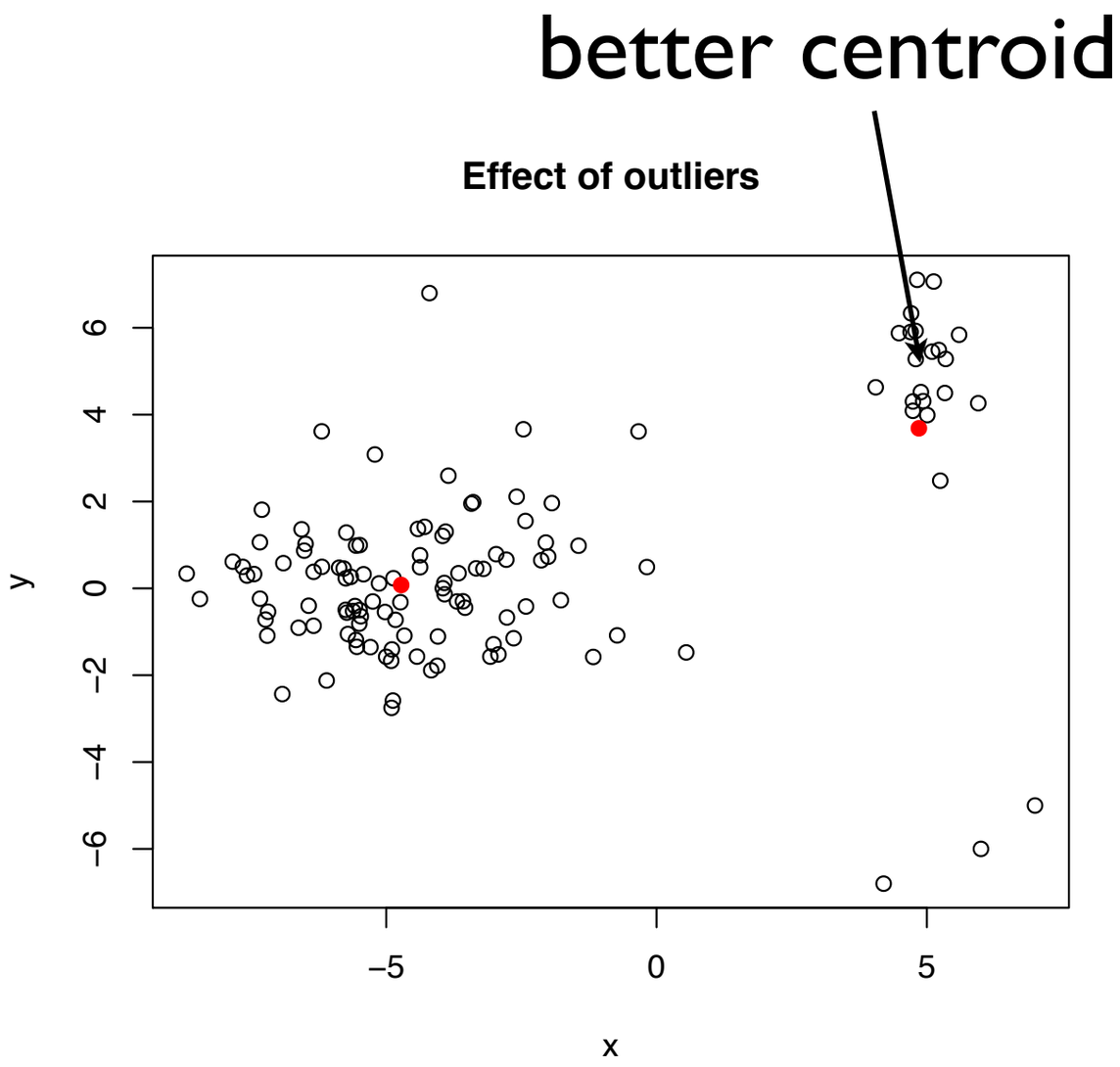
Centroid initialization

- Important for quick convergence and quality
- Randomly select k points as centroids
  - Clustering fails if two centroids are in the same cluster
- k-means++ addresses this



Outliers

- Real data is messy
- Outliers can affect k-means centroids

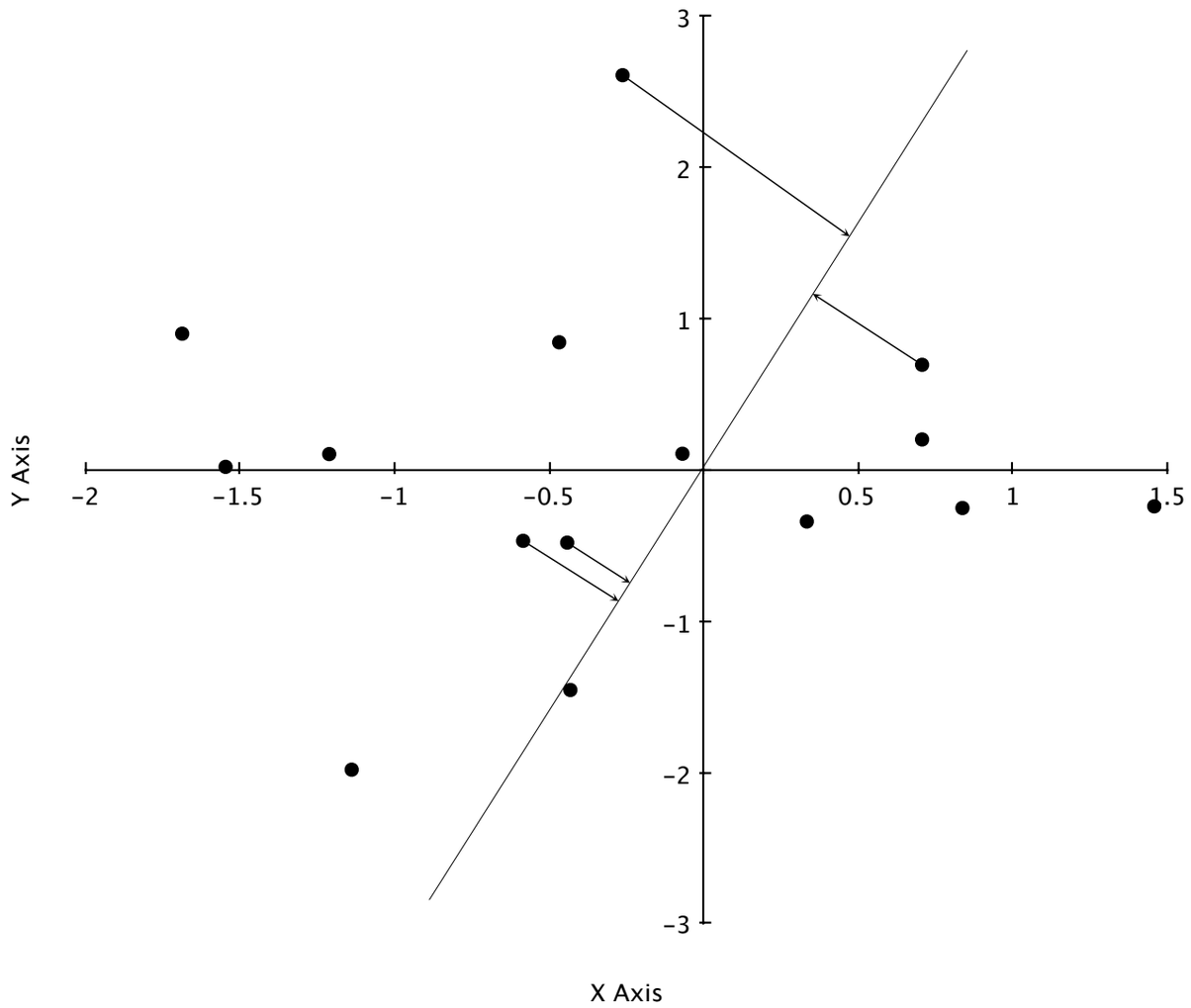


Closest cluster: reducing k

- Avoid computing the distance from a point to every cluster
  - Random Projections
    - projection search
    - locality-sensitive hashing search

Random Projections

- Unit length vectors with normally distributed components



Closest cluster: reducing d

- Principal Component Analysis
  - compute SVD
- Random Projections
  - multiply data by projection matrix

k-means as a MapReduce

- Can't split k-means loop directly
- Must express a single k-means step as a MapReduce

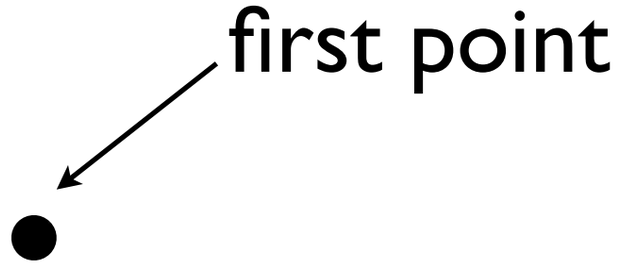
Now for something completely different

- *Fast and Accurate k-means For Large Datasets*
  M. Schindler, A. Wong, A. Meyerson

  - http://books.nips.cc/papers/files/nips24/NIPS2011_1271.pdf

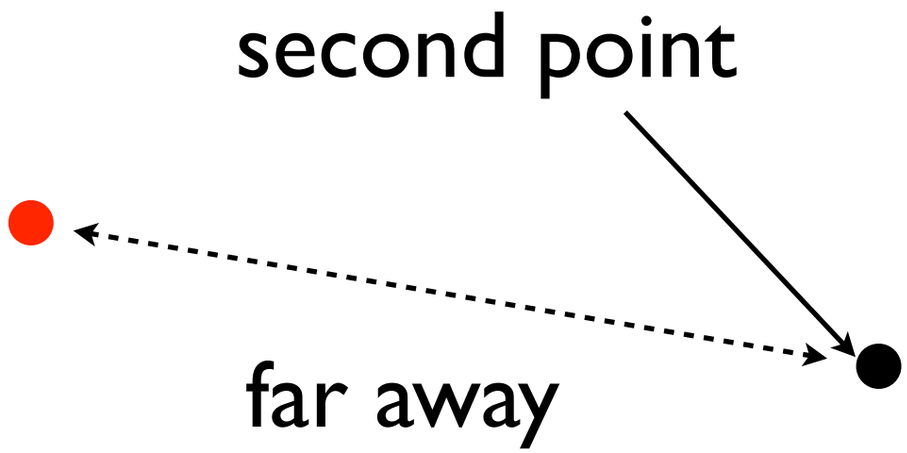
    - Attempt to build a "sketch" of the data in one pass
    - O(k log n) intermediate clusters
    - Can fit into memory
    - Ball k-means on the sketch for k final clusters

streaming k-means

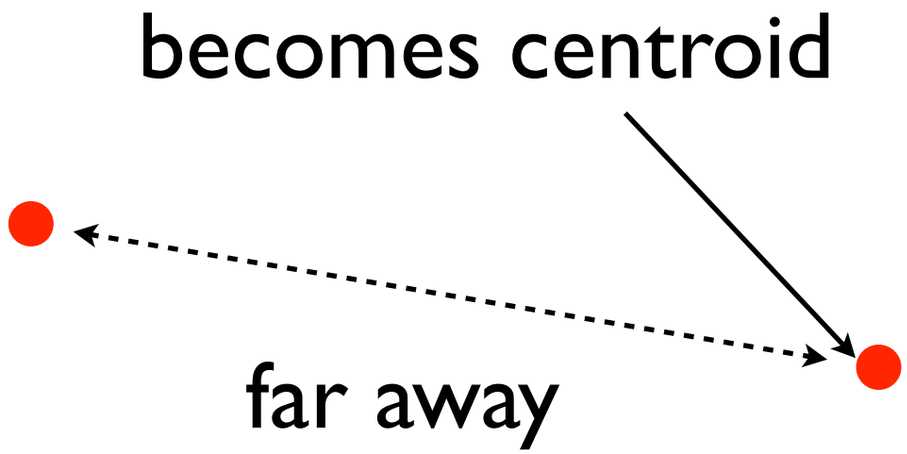


streaming k-means

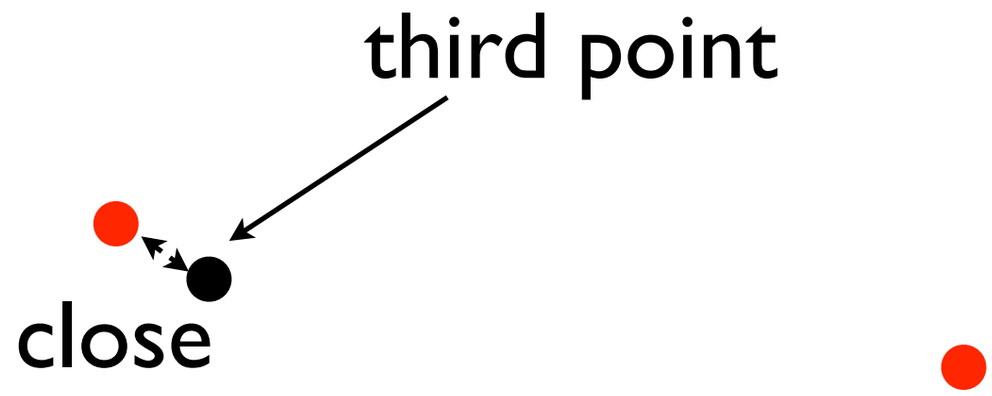
streaming k-means



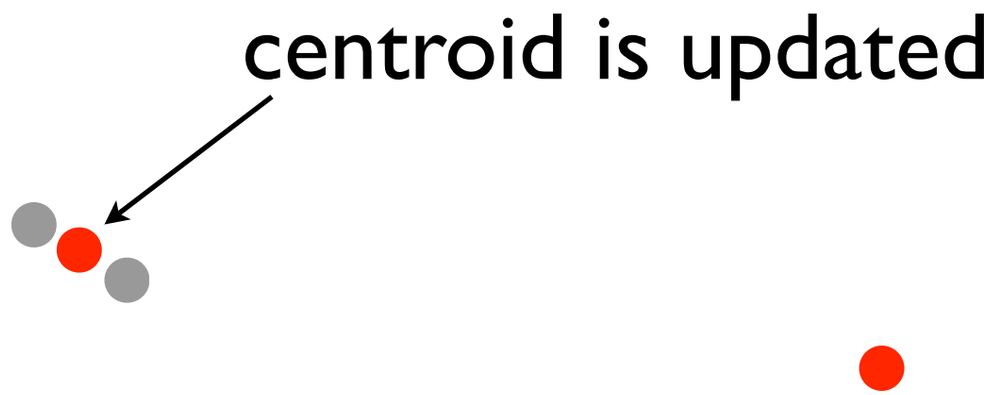
streaming k-means



streaming k-means



streaming k-means



streaming k-means

close
but becomes a
new centroid

streaming k-means

streaming k-means

- for each point **p**, with weight **w**
  - find the closest centroid to **p**, call it **c** and let d be the distance between **p** and **c**
  - if an event with probability proportional to **d** * **w** / **distanceCutoff** occurs
    - create a new cluster with **p** as its centroid
    - else, merge **p** into **c**
  - if there are too many clusters, increase **distanceCutoff** and cluster recursively

The big picture
MapReduce

- Can cluster all the points with just 1 MapReduce:
  - **m** mappers run streaming k-means
  - **1** reducer runs ball k-means to get k clusters

The big picture
Storm

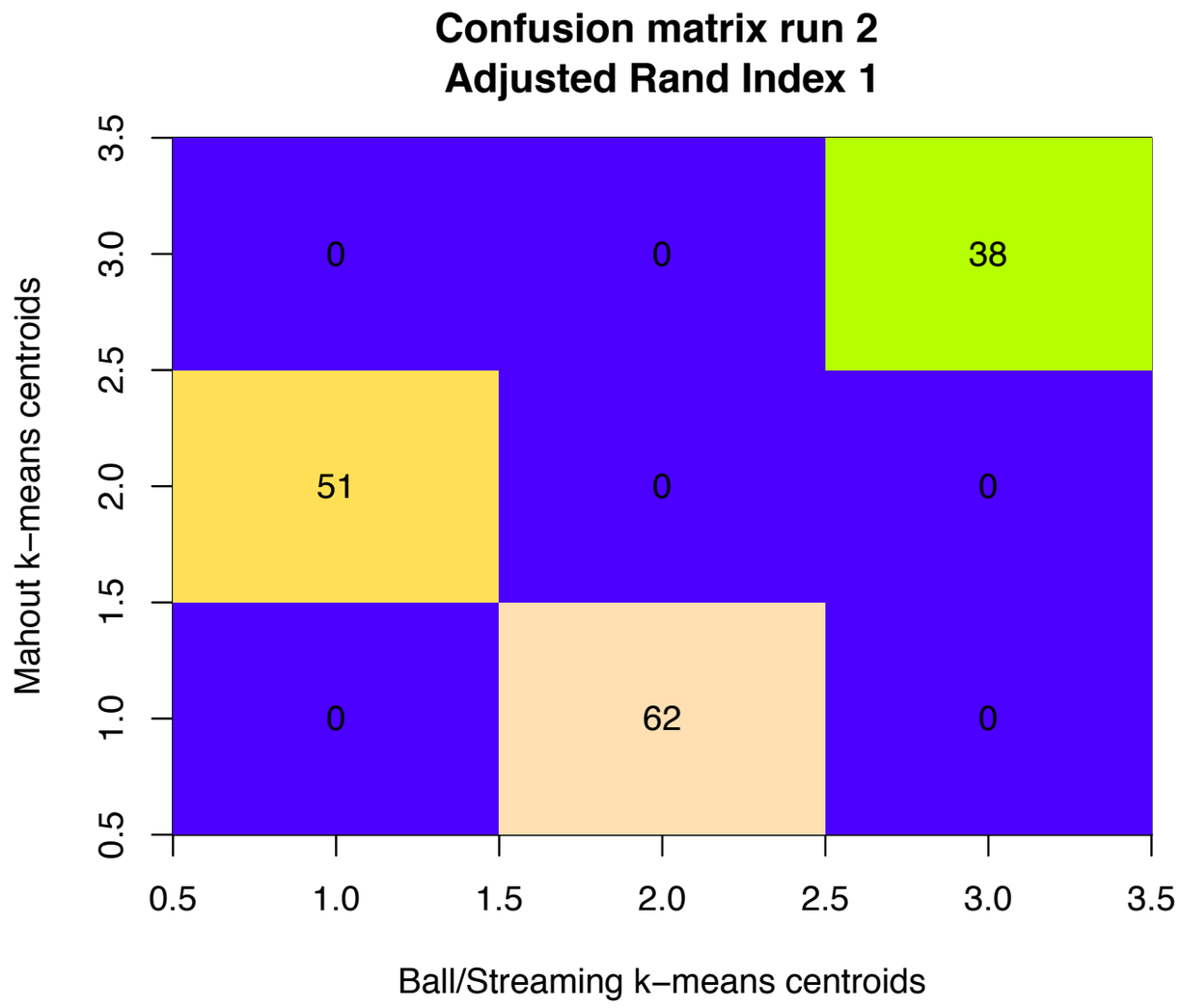
- Storm
  - streaming k-means bolt
    - Release the sketch when notified (e.g. tick tuples)
- Trident
  - streaming k-means partition aggregator
  - ball k-means aggregator

Results

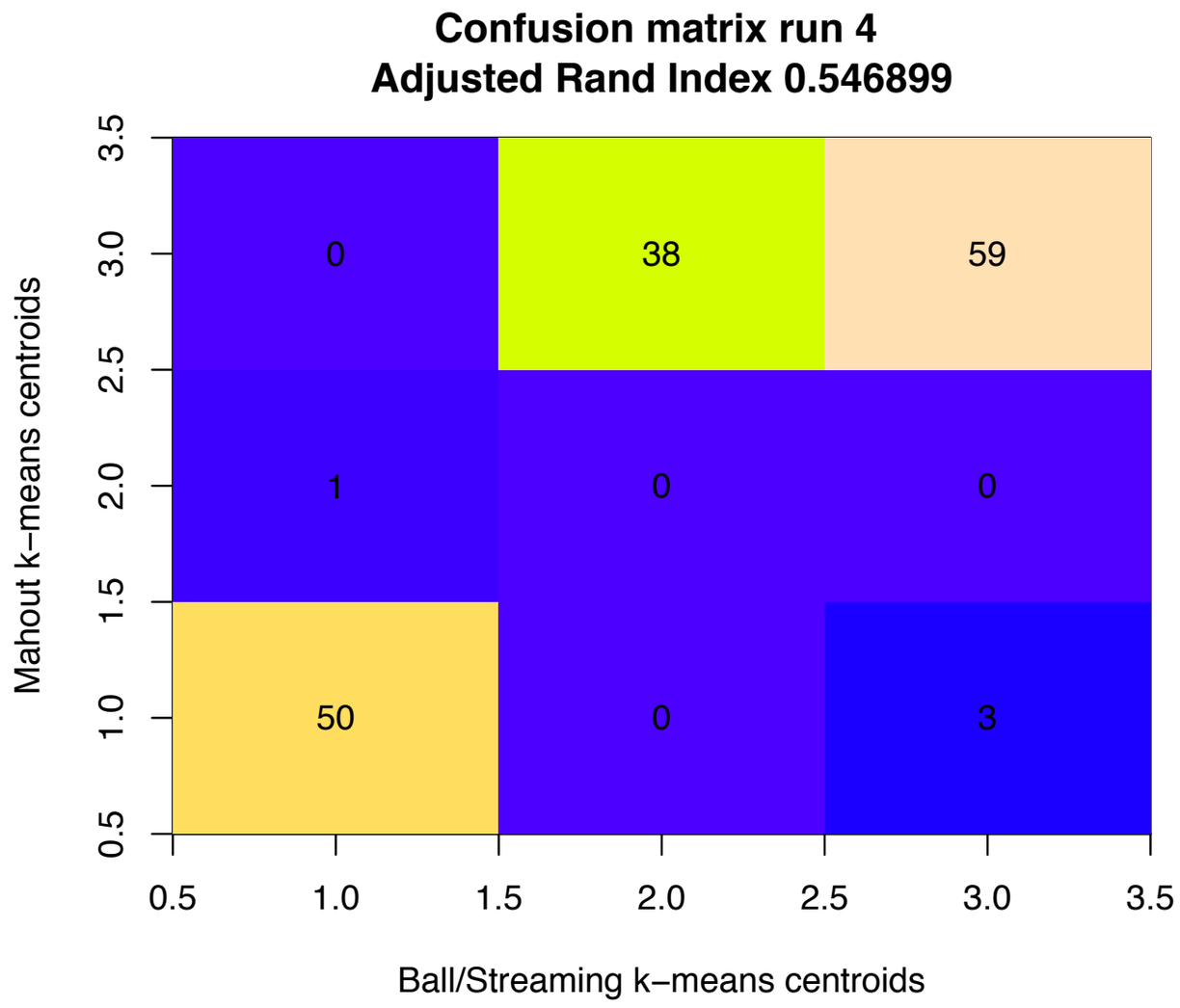
Quality

- Compared the quality on various small-medium UCI data sets
  - iris, seeds, movement, control, power
- Computed the following quality measures:
  - Dunn Index (higher is better)
  - Davies-Bouldin Index (lower is better)
  - Adjusted Rand Index (higher is better)
  - Total cost (lower is better)

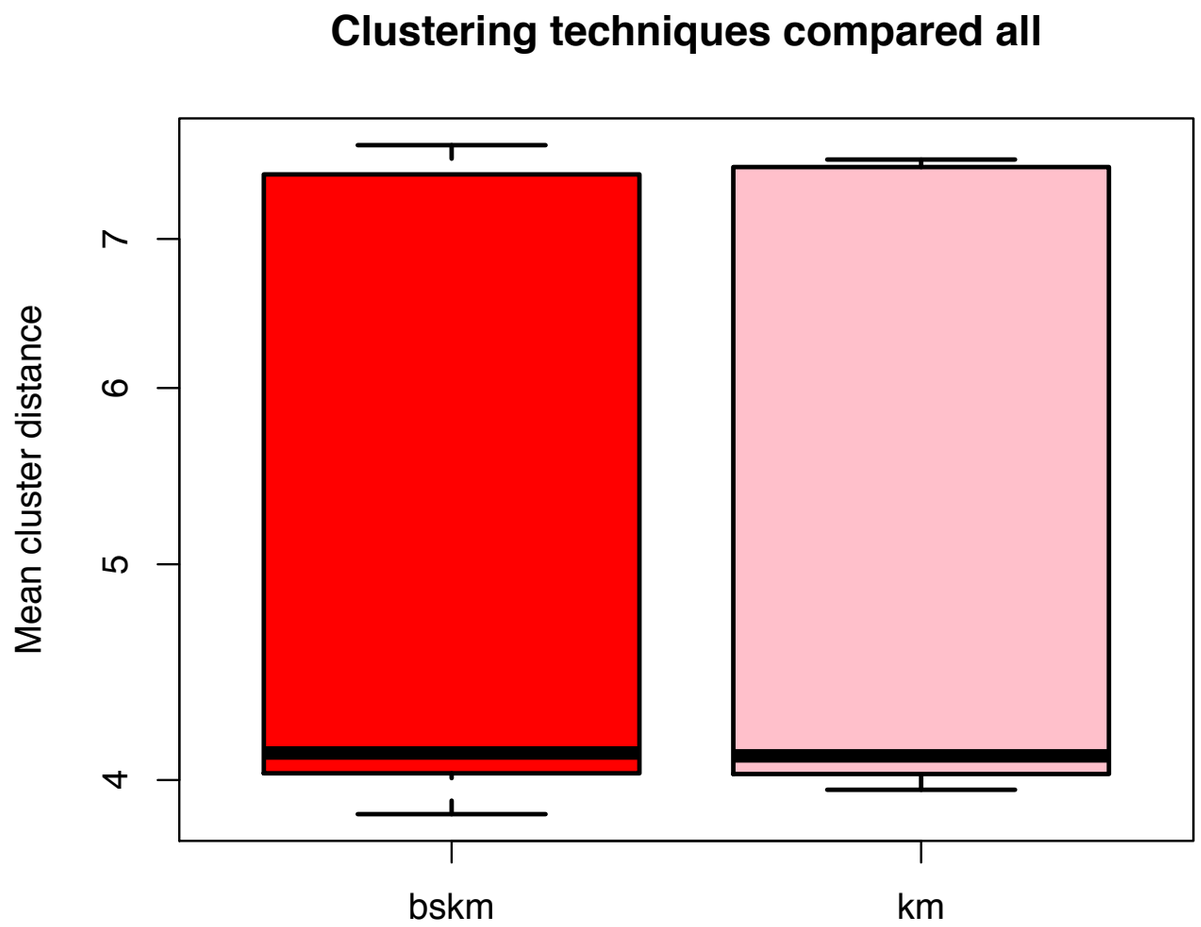
iris randplot-2



iris randplot-4

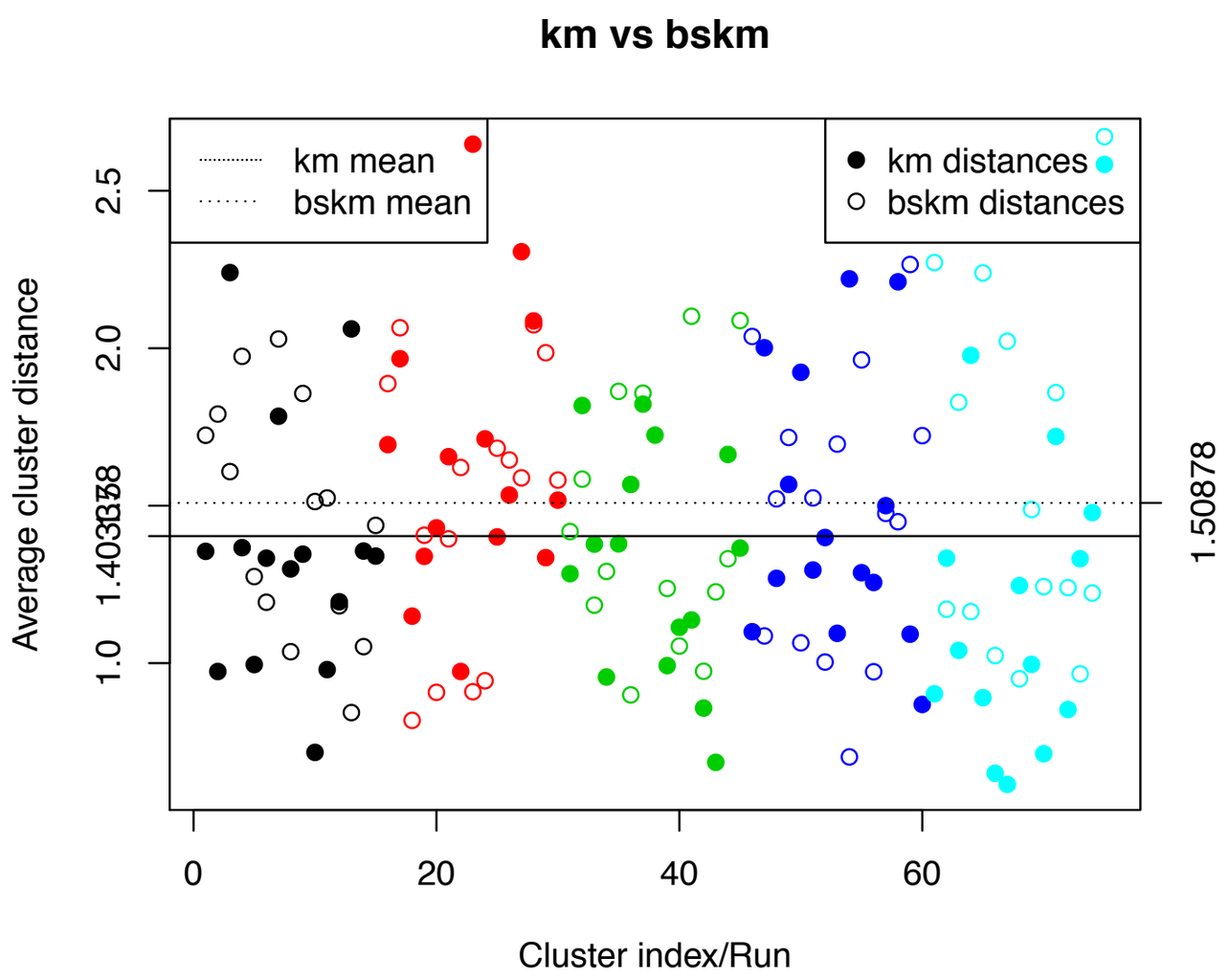


seeds compareplot

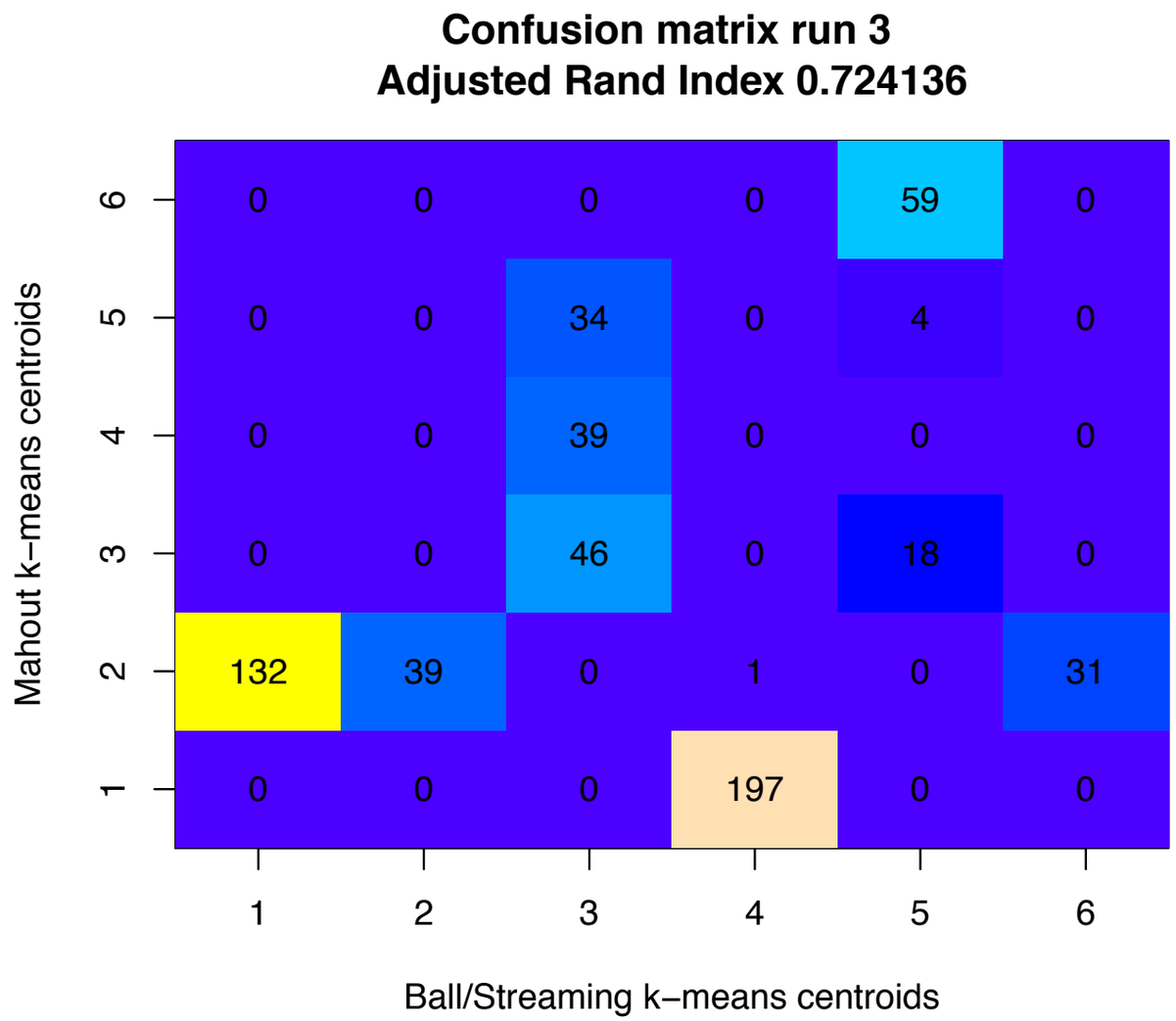


4.3116448 4.22126213333333

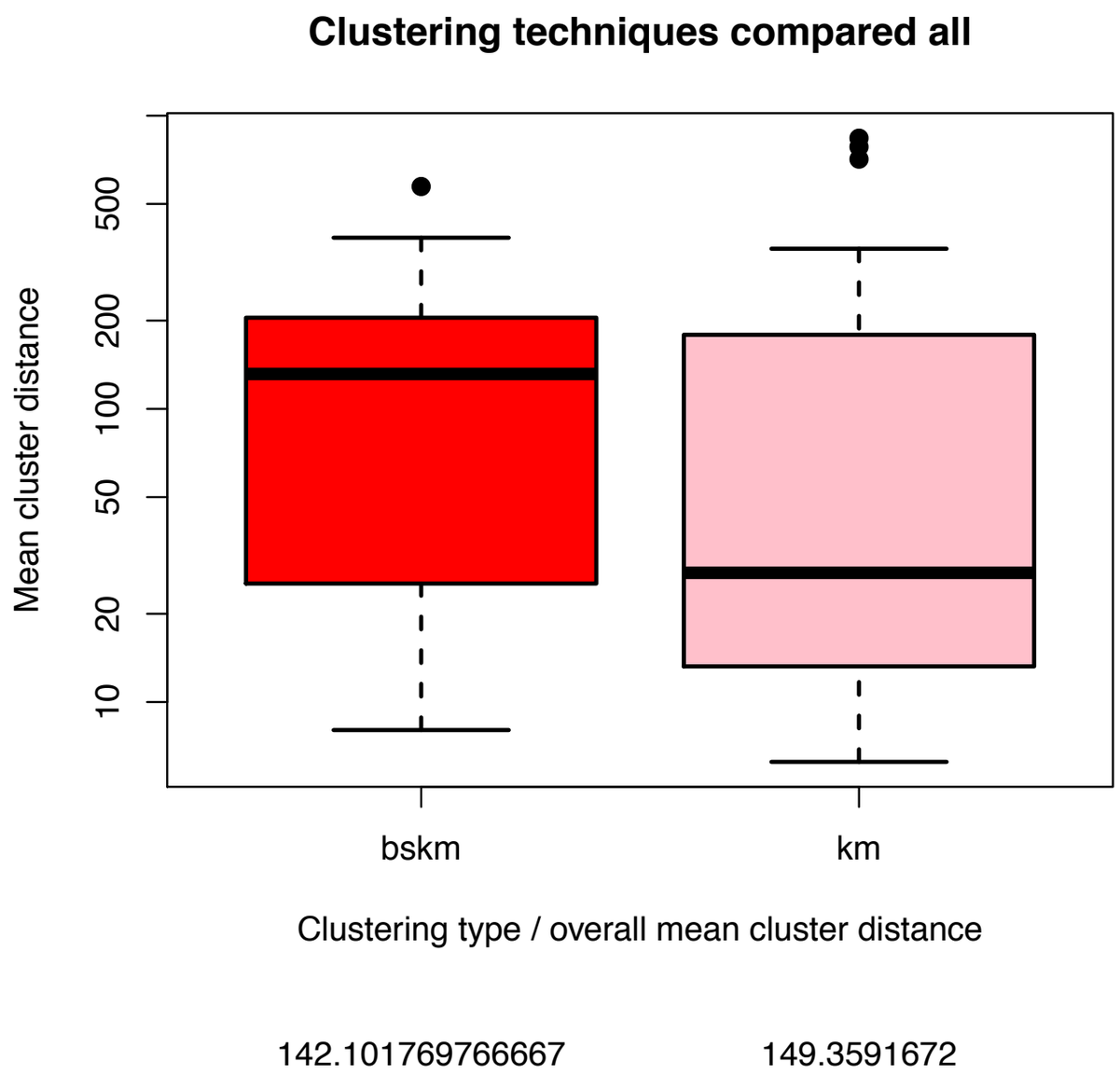
movement compareplot



control randplot-3

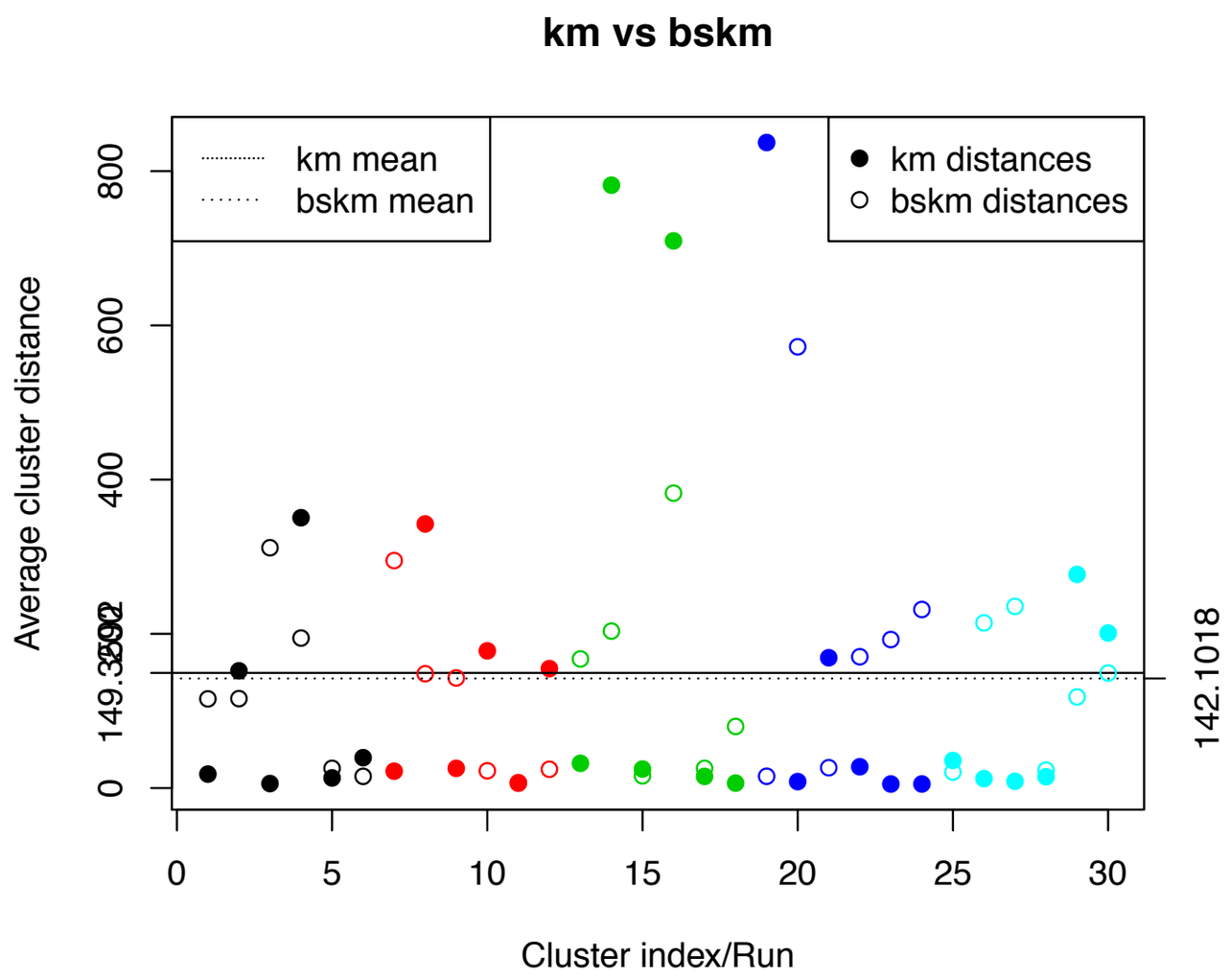


power allplot



142.101769766667 149.3591672

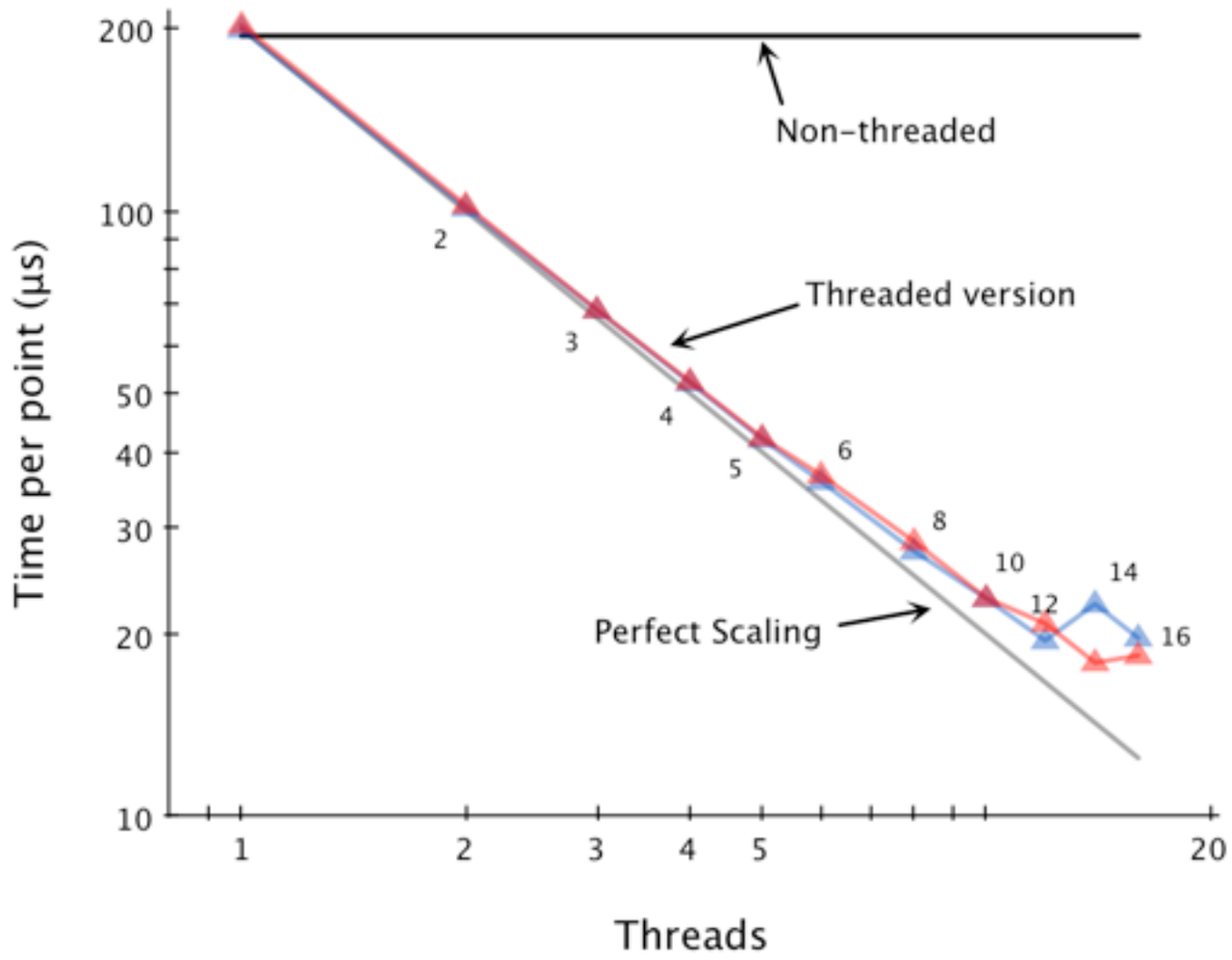
power compareplot



Overall (avg. 5 runs)

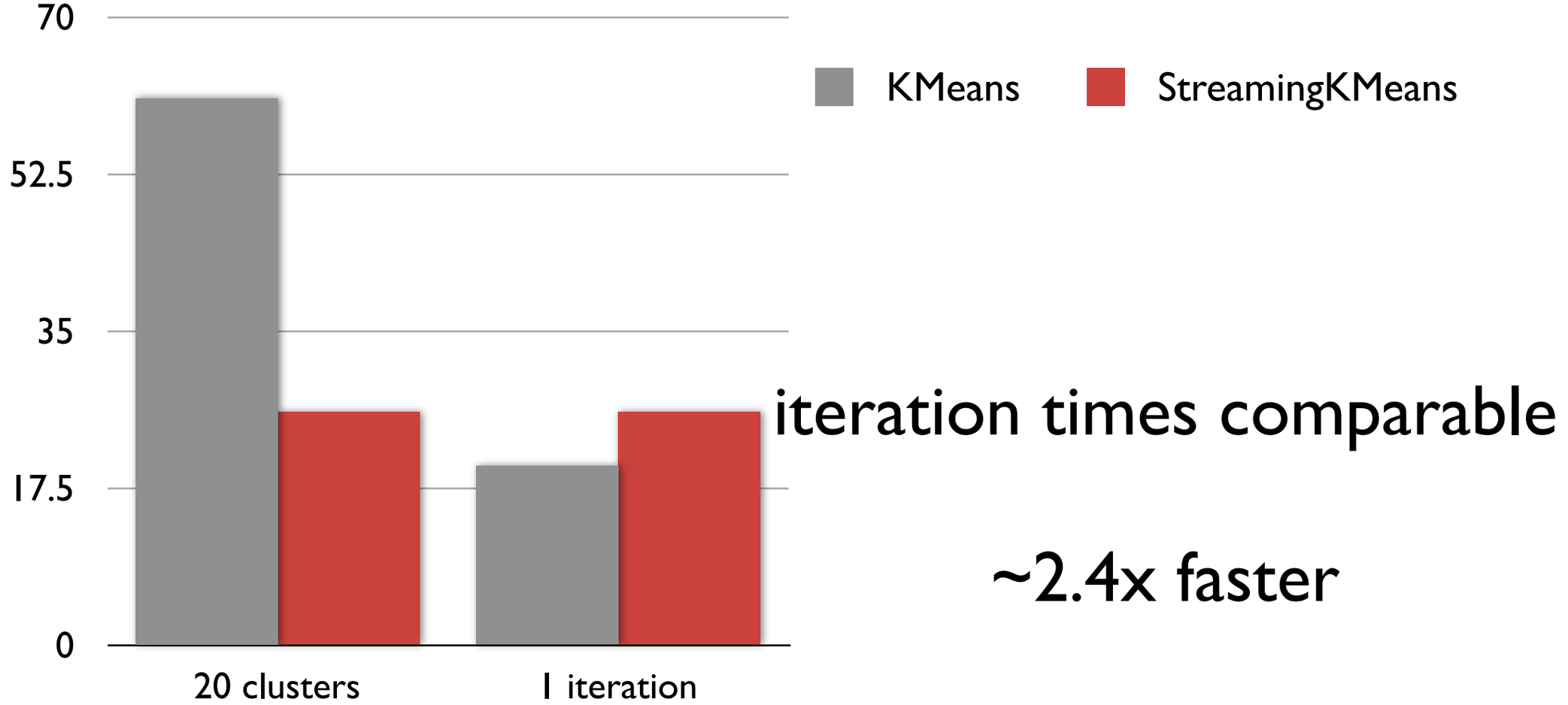
Dataset	Clustering	Avg. Dunn	Avg. DB	Avg. Cost	Avg. ARI
iris	km	**9.161**	**0.265**	124.146	0.905
	bskm	6.454	0.336	**117.859**	
seeds	km	**7.432**	**0.453**	**909.875**	0.980
	bskm	6.886	0.505	916.511	
movement	km	**0.457**	**1.843**	**336.456**	0.650
	bskm	0.436	2.003	347.078	
control	km	0.553	1.700	1014313	0.630
	bskm	**0.753**	**1.434**	**1004917**	
power	km	0.107	1.380	73339083	0.605
	bskm	**1.953**	**1.080**	**54422758**	

Speed (Threaded)



Speed (MapReduce)

Cluster Iterator runni...	mapr	20:46:11 05/29/2...	100%	100%	20m 42.4s	...070117_0142	20:46:11 05/29/2...
Cluster Iterator runni...	mapr	20:25:13 05/29/2...	100%	100%	20m 56.1s	...070117_0141	20:25:13 05/29/2...
Cluster Iterator runni...	mapr	20:03:58 05/29/2...	100%	100%	21m 12.5s	...070117_0140	20:03:58 05/29/2...
StreamingKMeansDri...	mapr	18:56:13 05/29/2...	100%	100%	25m 30.1s	...070117_0139	18:56:13 05/29/2...

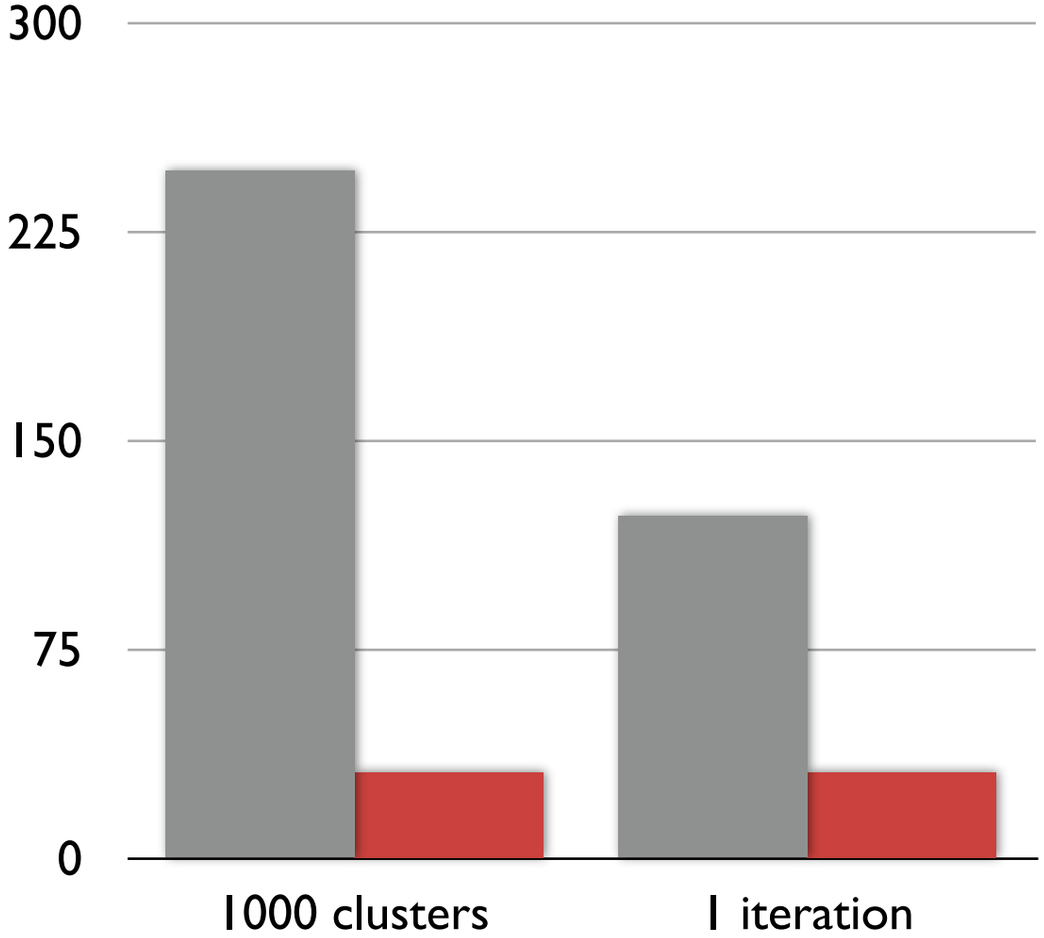


iteration times comparable

~2.4x faster

Speed (MapReduce)

Cluster Iterator runni…	mapr	23:51:45 05/25/2…	100%	100%	2h 3.1m	…070117_0116	23:51:45 05/25/2…
Cluster Iterator runni…	mapr	21:47:27 05/25/2…	100%	100%	2h 4.3m	…070117_0115	21:47:27 05/25/2…
StreamingKMeansDri…	mapr	20:43:47 05/25/2…	100%	100%	30m 12.2s	…070117_0114	20:43:47 05/25/2…



benefits from approximate nearest-neighbor search

~8x faster

Code

Clustering algorithms	`BallKMeans` `StreamingKMeans`
Fast nearest-neighbor search	`ProjectionSearch`
Quality metrics	`ClusteringUtils`
MapReduce classes	`StreamingKMeansMapper` `StreamingKMeansReducer`
Storm classes	`StreamingKMeansBolt`

- Now available in Apache Mahout trunk
- Prototype for Storm
  http://github.com/dfilimon/streaming-storm

Thank you!
Questions?

dfilimon@apache.org
dangeorge.filimon@gmail.com