CS 4602

Introduction to Machine Learning

Clustering

Instructor: Po-Chih Kuo

Roadmap

- Introduction and Basic Concepts
- Regression
- Bayesian Classifiers
- Decision Trees
- Linear Classifier
- Neural Networks
- Deep learning
- Convolutional Neural Networks
- The others
- KNN
- Clustering
- Data Exploration & Dimensionality reduction
- Model Selection and Evaluation

Outline

- Motivation
- Choosing (dis)similarity measures – **a critical step in clustering**
- Clustering algorithms

Ref: Slides from Georg Gerber

What is clustering?

- A way of grouping together data samples that are ***similar*** according to some criteria
- A form of ***unsupervised learning*** – you don’t have examples (testing data) demonstrating how the data should be grouped together
- It’s a method of ***EDA (exploratory data analysis)***– a way of looking for patterns or structure in the data that are of interest

Some applications

- Streaming Services
  - to identify high usage and low usage users so that they can know who they should spend most of their advertising dollars on.

Some applications

- **Sports Science**
  - To identify players that are similar to each other so that they can have these players practice with each other and perform specific drills based on their strengths and weaknesses.

12/11/2022

POINTS

1.	**Joel Embiid** PHI	**53**
2.	Bojan Bogdanovic DET	38
3.	LeBron James LAL	35
3.	Zion Williamson NOP	35
5.	Anthony Davis LAL	34

REBOUNDS

1.	**Giannis Antetokounmpo** MIL	**18**
2.	Anthony Davis LAL	15
2.	Bobby Portis MIL	15
4.	Clint Capela ATL	14
5.	DeMar DeRozan CHI	13

ASSISTS

1.	**James Harden** PHI	**16**
2.	Trae Young ATL	14
3.	Chris Paul PHX	11
4.	Killian Hayes DET	9
4.	Russell Westbrook LAL	9

BLOCKS

1.	**Bol Bol** ORL	**3**
1.	Derrick Jones Jr. CHI	3
1.	Brook Lopez MIL	3
4.	Giannis Antetokounmpo MIL	2
4.	Mo Bamba ORL	2

STEALS

1.	**Andre Drummond** CHI	**5**
2.	Larry Nance Jr. NOP	4
2.	Fred VanVleet TOR	4
4.	RJ Barrett NYK	3
4.	Tobias Harris PHI	3

TURNOVERS

1.	**Zach LaVine** CHI	**7**
2.	Trae Young ATL	6
3.	Jalen Brunson NYK	5
3.	Markelle Fultz ORL	5
3.	Jrue Holiday MIL	5

THREE POINTERS MADE

1.	**Bojan Bogdanovic** DET	**6**
1.	Bogdan Bogdanovic ATL	6
3.	Mikal Bridges PHX	5
3.	Terry Rozier CHA	5
5.	Saddiq Bey DET	4

FREE THROWS MADE

1.	**DeMar DeRozan** CHI	**14**
2.	Joel Embiid PHI	11
3.	Anthony Davis LAL	10
3.	Fred VanVleet TOR	10
5.	Bojan Bogdanovic DET	8

FANTASY POINTS

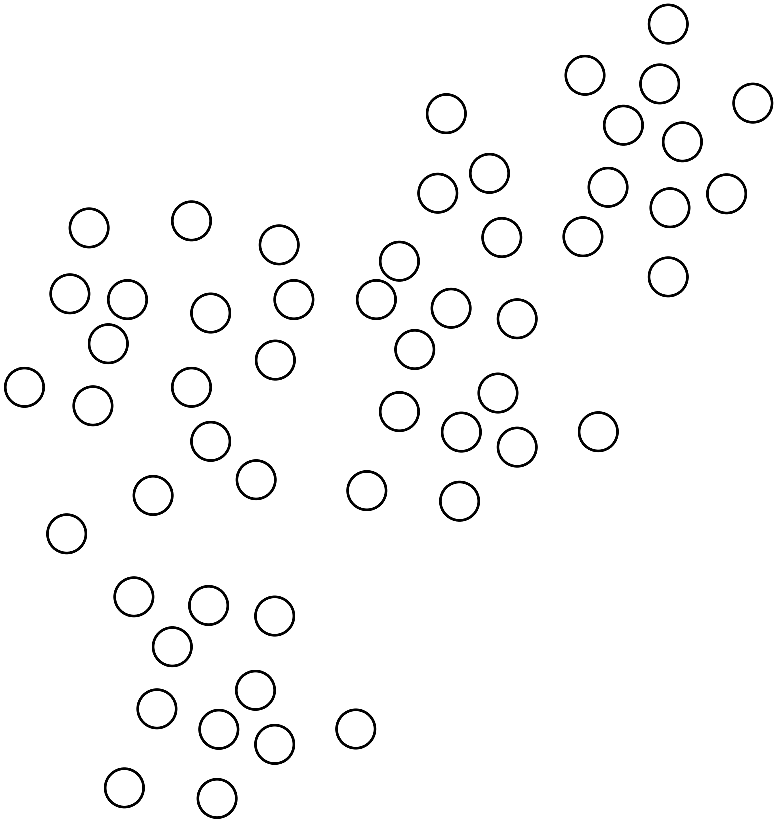
1.	**Joel Embiid** PHI	**72.9**
2.	Anthony Davis LAL	70.5
3.	DeMar DeRozan CHI	60.6
4.	RJ Barrett NYK	58.8
5.	James Harden PHI	52.8

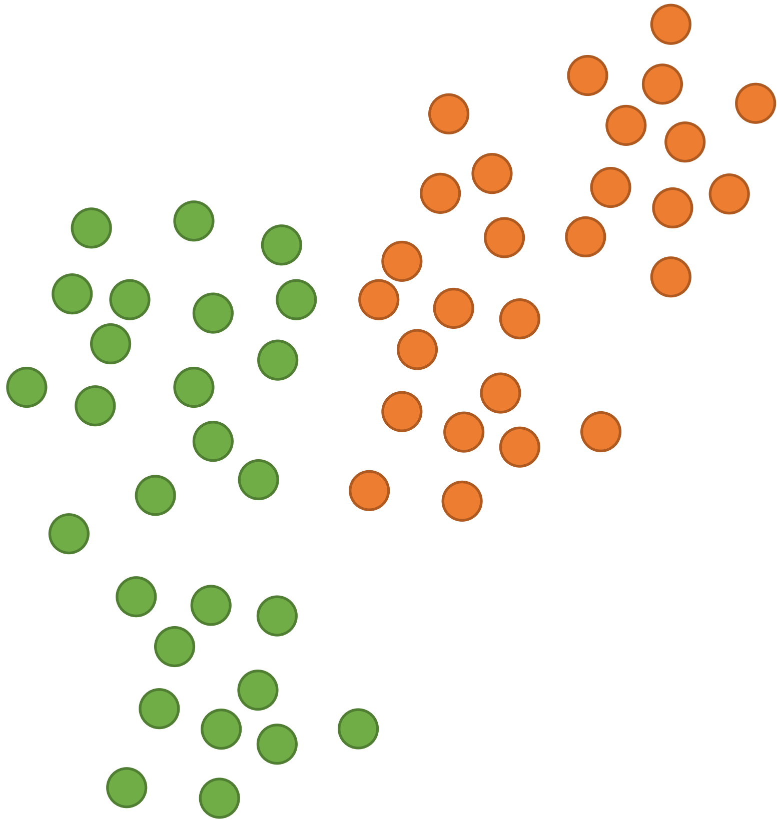
by features

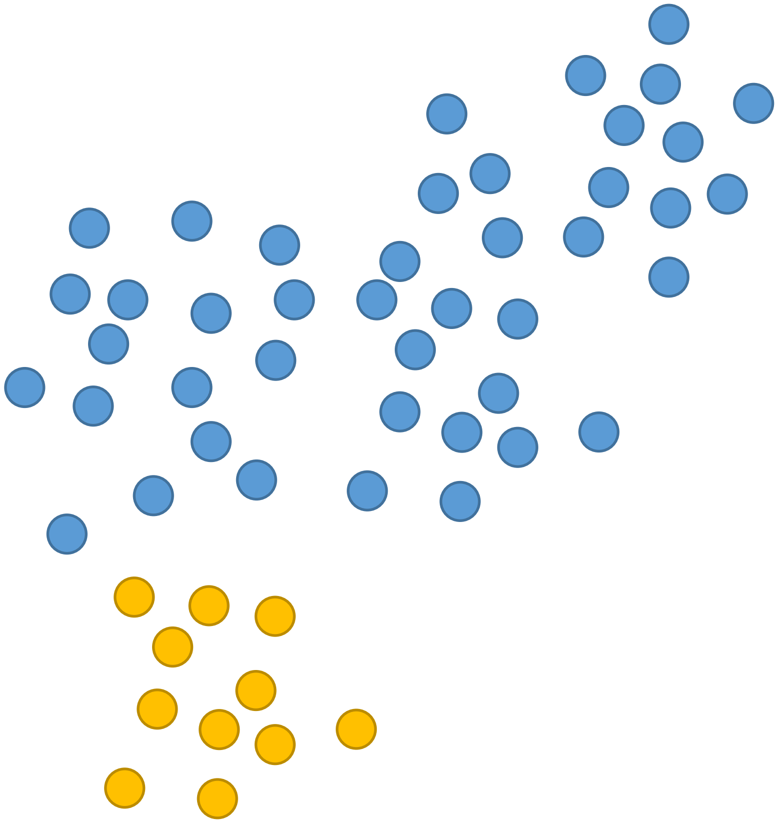
Example	Attributes										Ta[redacted]get
	Alt.	Bar	Fri	Hun	Pat	Price	Rain	Res	Type	Est.	W[redacted]it
X_1	T	F	F	T	Some	$$$	F	T	French	0-10	[redacted]
X_2	T	F	F	T	Full	$	F	F	Thai	30-60	[redacted]
X_3	F	T	F	F	Some	$	F	F	Burger	0-10	[redacted]
X_4	T	F	T	T	Full	$	F	F	Thai	10-30	[redacted]
X_5	T	F	T	F	Full	$$$	F	T	French	>60	[redacted]
X_6	F	T	F	T	Some	$$	T	T	Italian	0-10	[redacted]
X_7	F	T	F	F	None	$	T	F	Burger	0-10	[redacted]
X_8	F	F	F	T	Some	$$	T	T	Thai	0-10	[redacted]
X_9	F	T	T	F	Full	$	T	F	Burger	>60	[redacted]
X_{10}	T	T	T	T	Full	$$$	F	T	Italian	10-30	[redacted]
X_{11}	F	F	F	F	None	$	F	F	Thai	0-10	[redacted]
X_{12}	T	T	T	T	Full	$	F	F	Burger	30-60	[redacted]

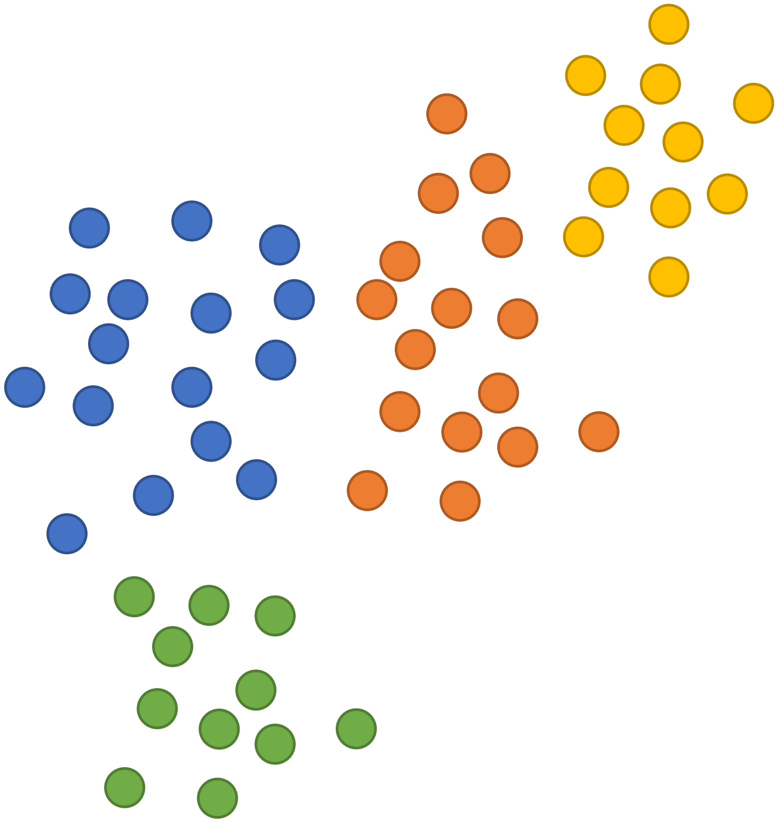
by samples

How to cluster the data?







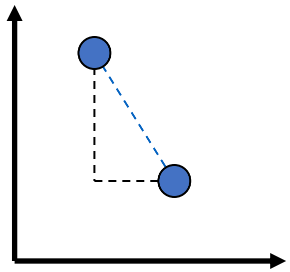


There is no single right answer!

How do we define “similarity”?

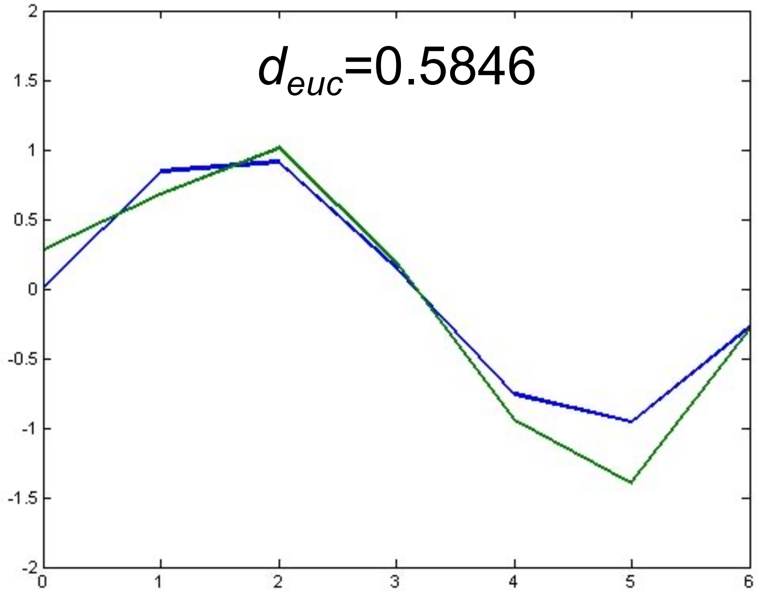
- The goal is to group together “**similar**” data – but what does this mean?
- No single answer – it depends on what we want to find or emphasize in the data;
  - Clustering is an “art”!
- The similarity measure is often more important than the clustering algorithm used
- This is always a ***pair-wise*** measure

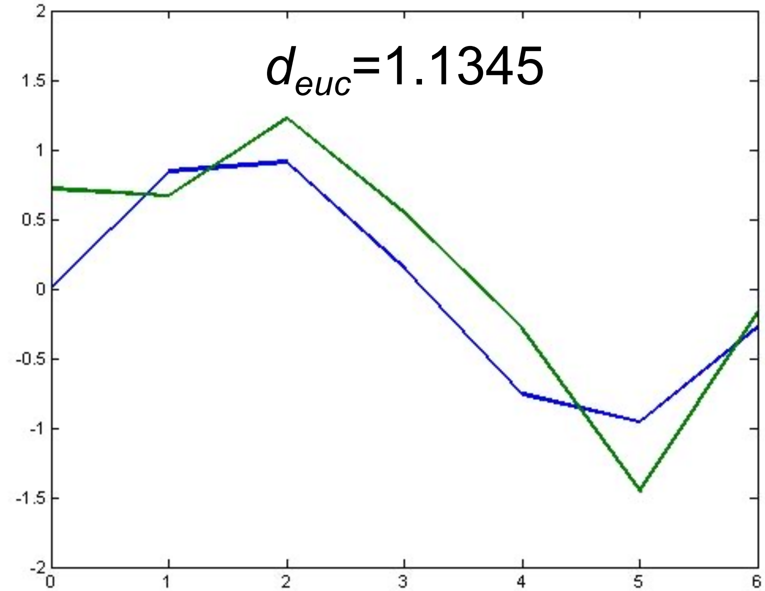
Euclidean distance

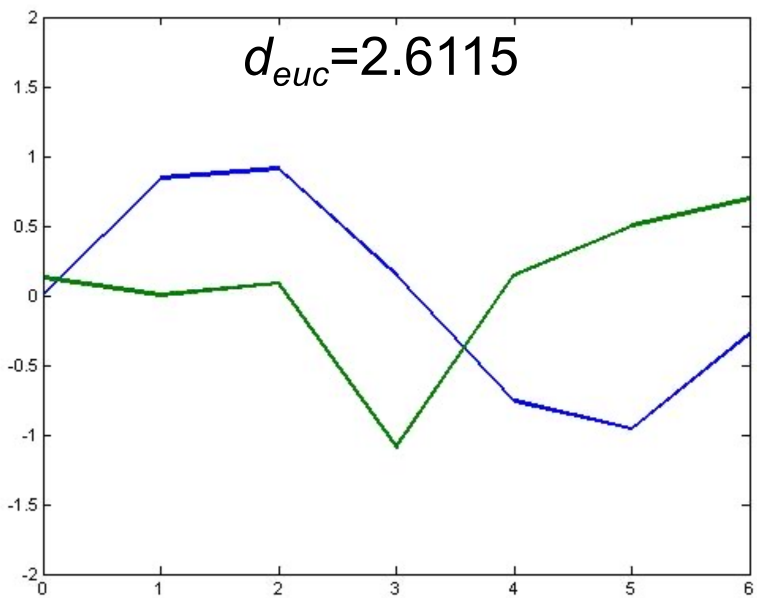


$$d_{euc}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$

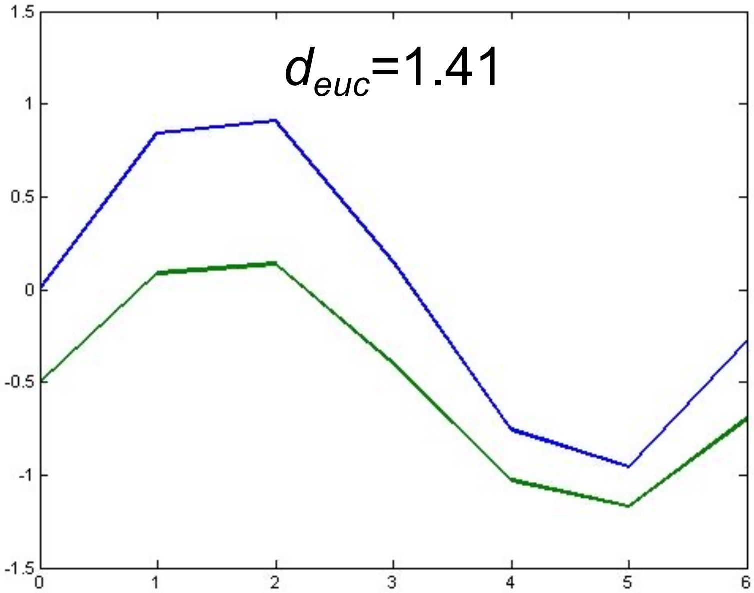
- Here *n* is the number of dimensions in the data vector. For instance:
  - Number of features (when clustering samples)
  - Number of samples (when clustering features)

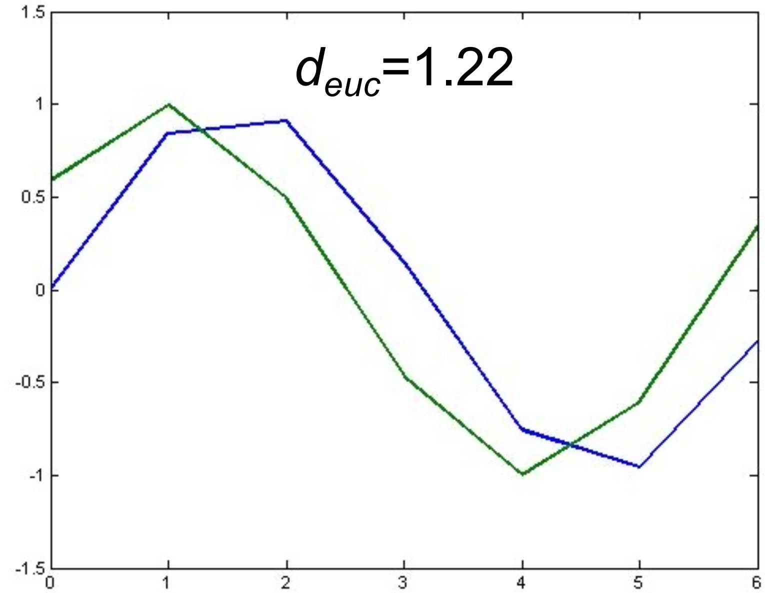






These examples of Euclidean distance match our intuition of dissimilarity well…



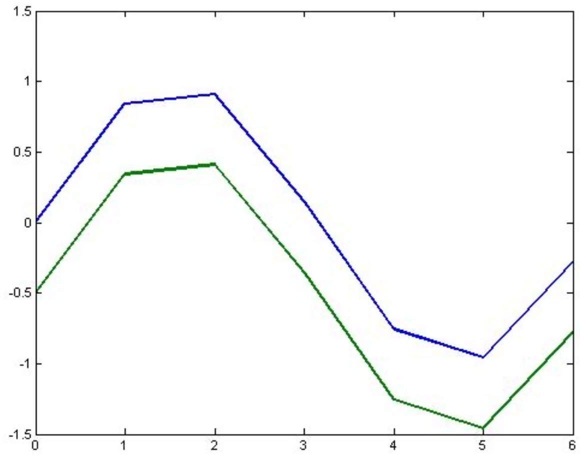


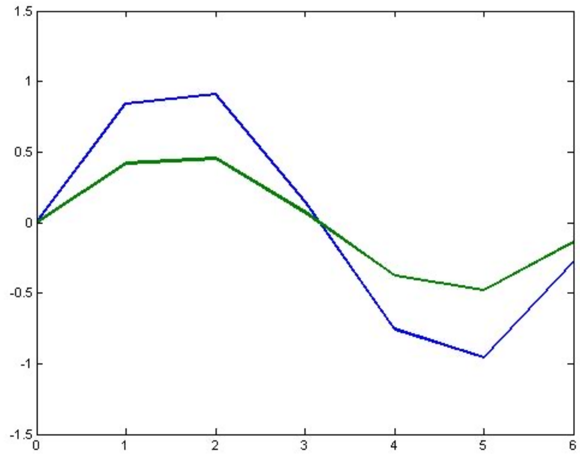
…But what about these?

What might be going on with the data profiles on the left? On the right?

Correlation

- We might care more about the overall shape of data profiles rather than the actual magnitudes
- We might want to consider samples similar when they are "up" and "down" together





Pearson Linear Correlation

$$\rho(\mathbf{x},\mathbf{y}) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

$$\bar{x} = \frac{1}{n}\sum_{i}^{n} x_i$$

$$\bar{y} = \frac{1}{n}\sum_{i}^{n} y_i$$

- We're shifting the data profiles down (subtracting the means) and scaling by the standard deviations (i.e., making the data have mean = 0 and std = 1)

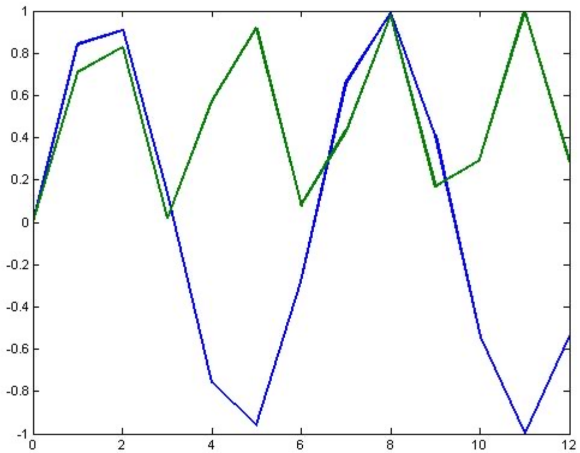
PLC (cont.)

- Pearson linear correlation (PLC) is a measure that is invariant to scaling and shifting (vertically) of the data values
- Always between –1 and +1 (perfectly anti-correlated and perfectly correlated)
- This is a similarity measure, but we can easily make it into a dissimilarity measure:

$$d_p = \frac{1 - \rho(\mathbf{x}, \mathbf{y})}{2}$$

PLC (cont.)

- PLC only measures the degree of a *linear* relationship between two profiles!
- If you want to measure other relationships, there are many other possible measures

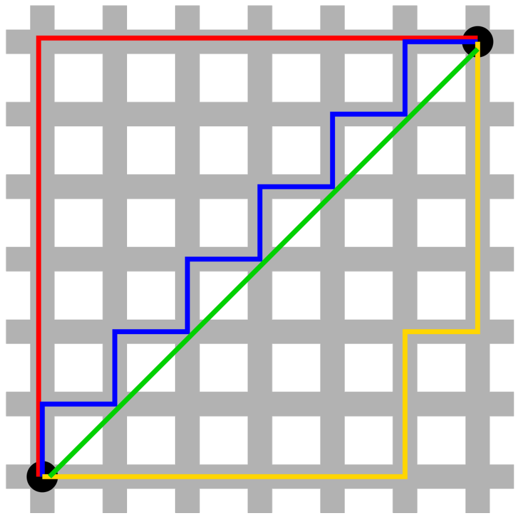


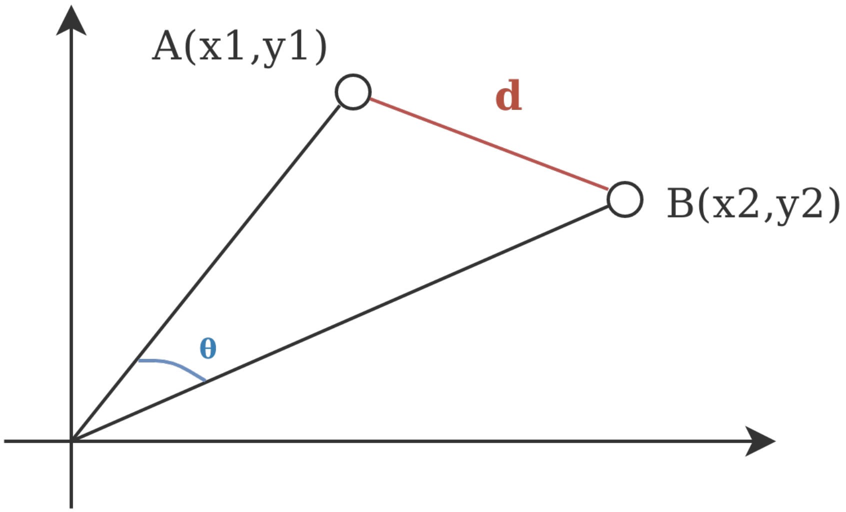
ρ = 0.0249, so d_p = 0.4876

The green curve is the square of the blue curve – this relationship is not captured with PLC

Other measures

- Manhattan distance (or Cityblock, or *l1*), cosine distance





Manhattan is preferred over Euclidean distance:

1. if the dataset has discrete attributes
2. for the case of high dimensional data.

Outline

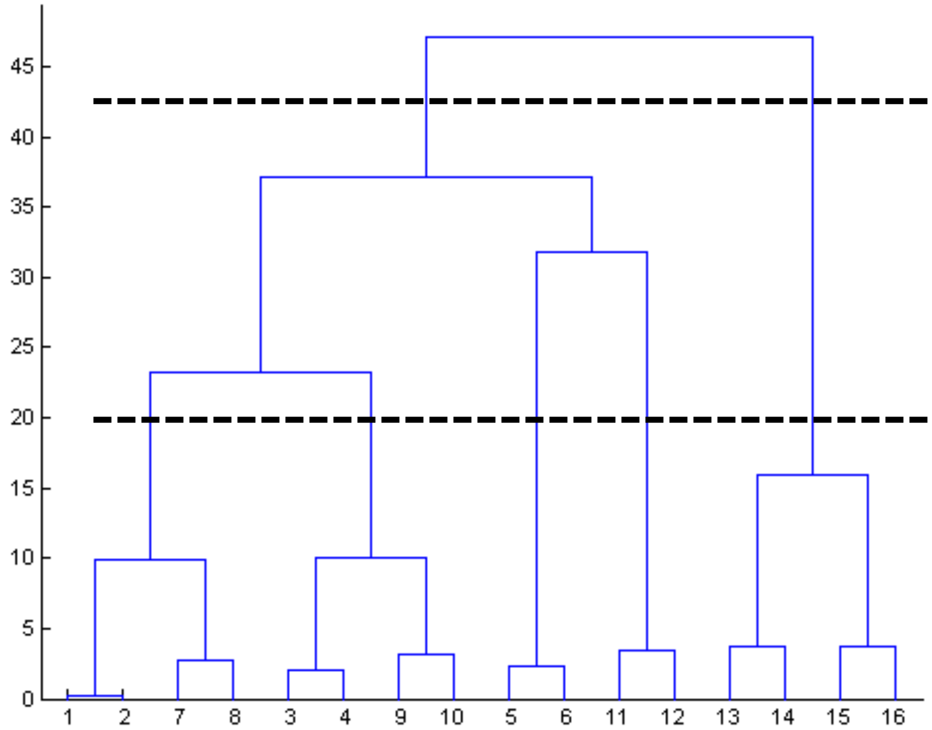
- Motivation
- Choosing (dis)similarity measures – **a critical step in clustering**
- Clustering algorithms
  - Hierarchical clustering
  - K-means

Ref: Slides from Georg Gerber

Hierarchical Clustering

- We start with every data point in a separate cluster
- We keep merging the most similar pairs of data points/clusters until we have one big cluster left
- This is called a bottom-up or agglomerative method

Hierarchical Clustering (cont.)



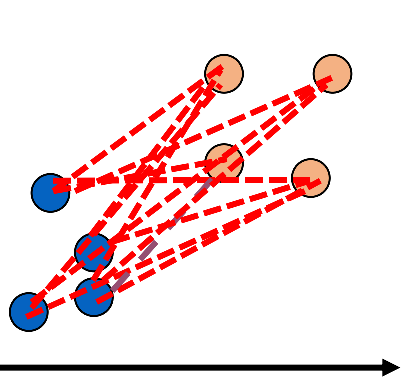
- This produces a binary tree or ***dendrogram***
- The final cluster is the root and each data item is a leaf
- The height of the bars indicate how close the items are

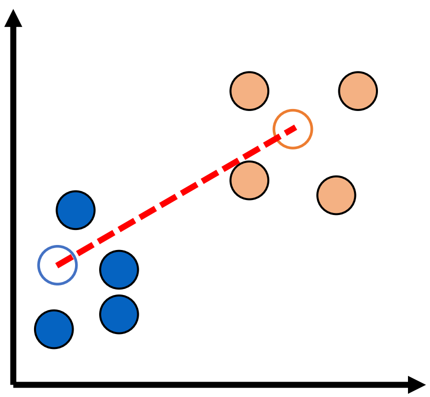
Linkage in Hierarchical Clustering

- We already know about distance measures between data items, but what about between a data item and a cluster or between two clusters?
- We just treat a data point as a cluster with a single item, so our only problem is to define a ***linkage*** method between clusters
- Again, there are lots of choices…

Average Linkage

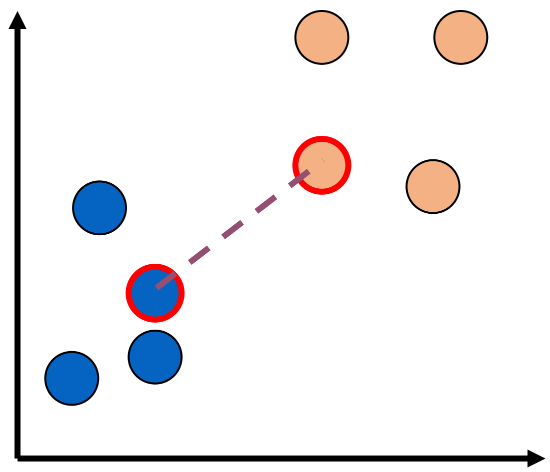
- Average linkage is defined as follows:
  - Each cluster c_i is associated with a mean vector μ_i which is the mean of all the data items in the cluster
  - The distance between two clusters c_i and c_j is then just $d(\mu_i, \mu_j)$
- This method is usually referred to as **centroid linkage** and **average linkage** is defined as the average of all pairwise distances between points in the two clusters





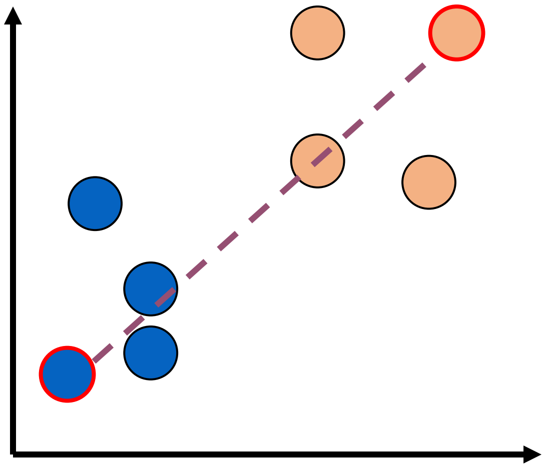
Single Linkage

- The minimum of all pairwise distances between points in the two clusters
- Tends to produce long, “loose” clusters



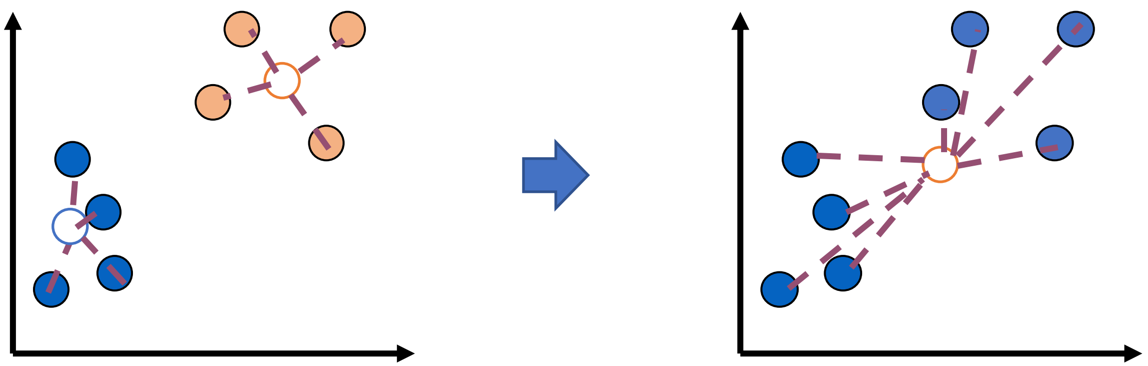
Complete Linkage

- The maximum of all pairwise distances between points in the two clusters
- Tends to produce very tight clusters

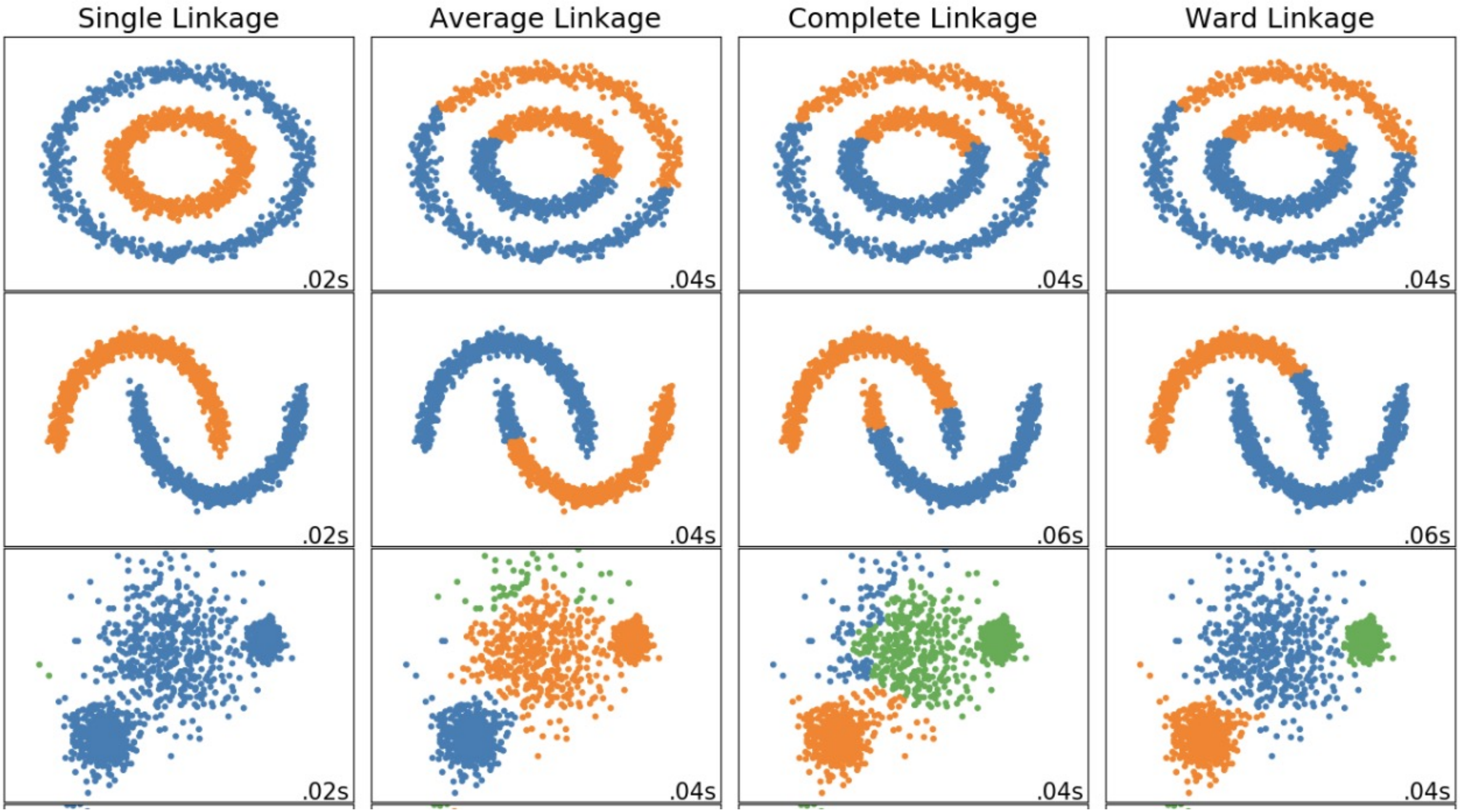


Ward's Method

- Consider merging two clusters, how does it change the total distance from centroids?

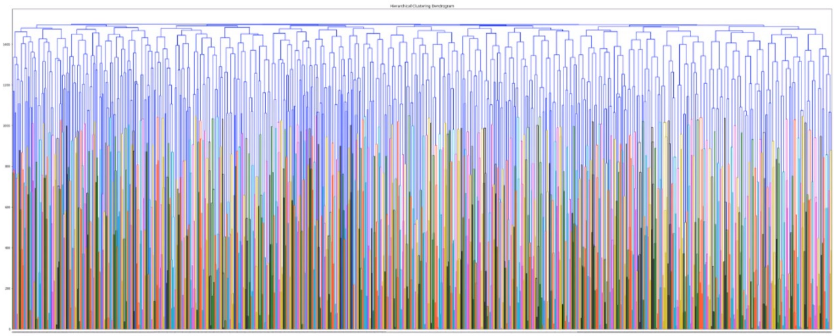


1. Find the centroid of each cluster.
2. Calculate the distance between each object and its cluster's centroid.
3. Calculate the sum of squared differences from Step 2.
4. Add up all the sums from Step 3.


Single Linkage
Average Linkage
Complete Linkage
Ward Linkage
.02s
.04s
.04s
.04s
.02s
.04s
.06s
.06s
.02s
.04s
.04s
.04s

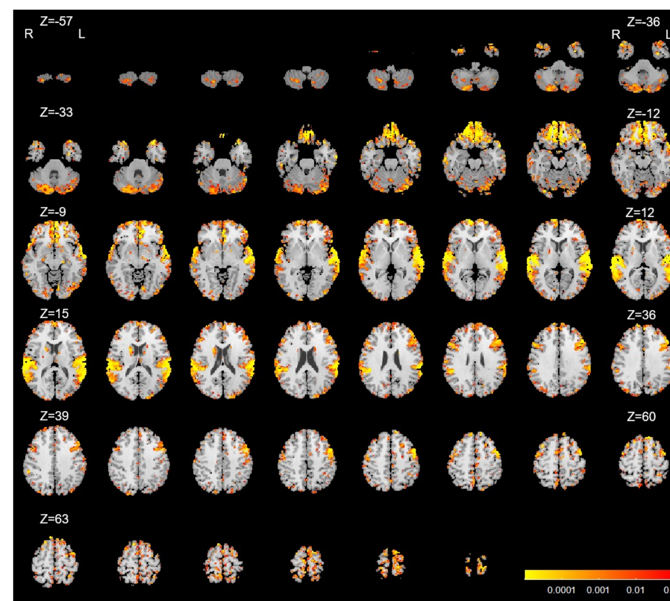
Hierarchical Clustering Issues

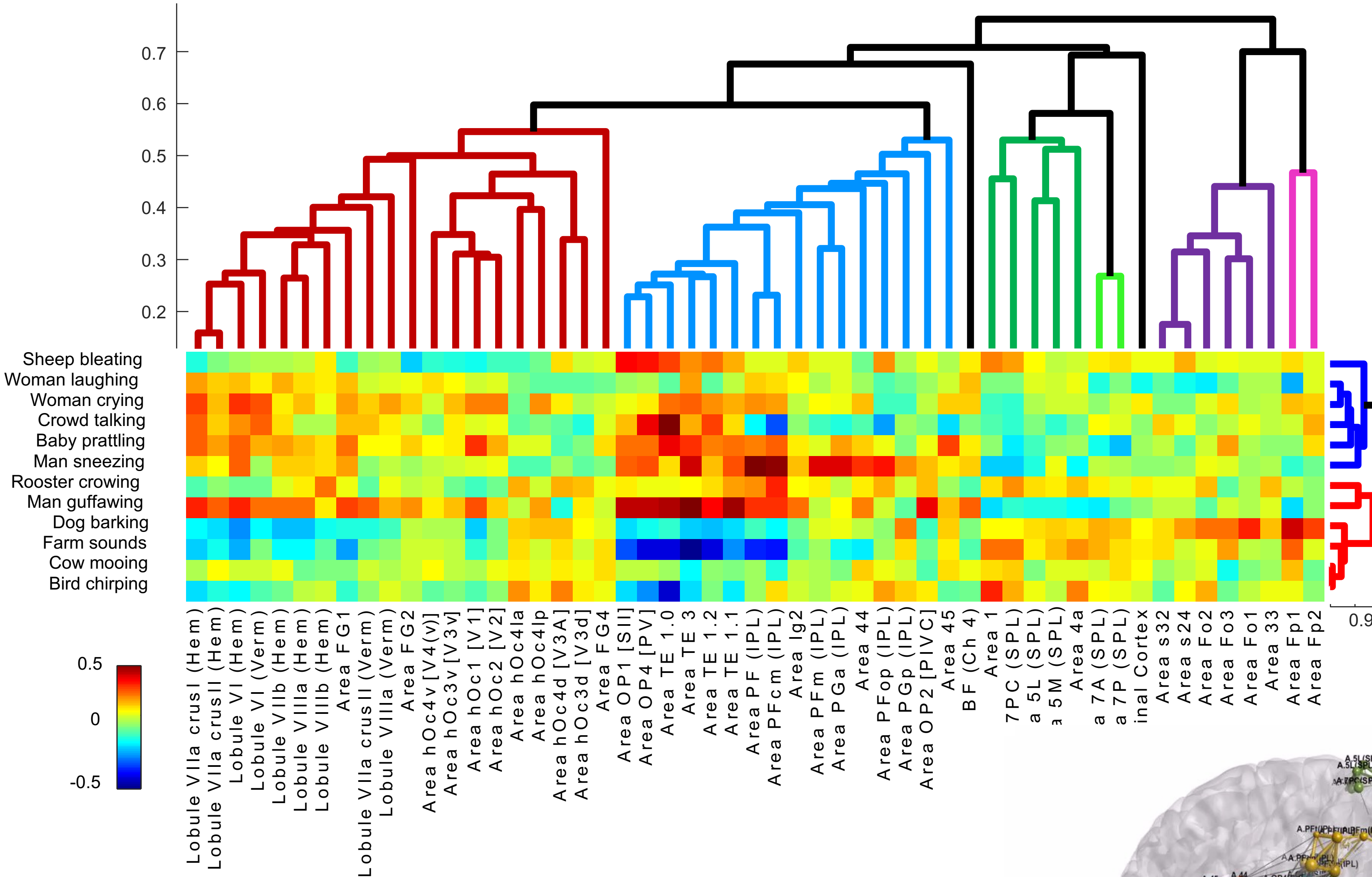
- Distinct clusters are not produced – sometimes this can be good, if the data has a hierarchical structure w/o clear boundaries (**No need to present the number of clusters**)
- There are methods for producing distinct clusters, but these usually involve specifying arbitrary **cutoff values**
- Heavy computation

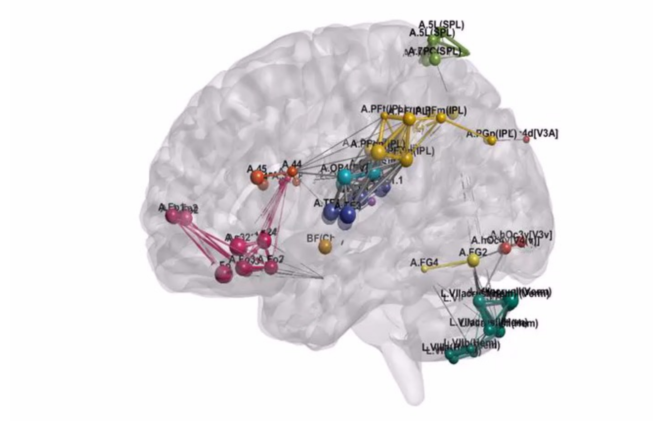


Example

- An fMRI experiment engaging long-term auditory stimulation reflects a real-world experience in the brain.




0.7
0.6
0.5
0.4
0.3
0.2
Sheep bleating
Woman laughing
Woman crying
Crowd talking
Baby prattling
Man sneezing
Rooster crowing
Man guffawing
Dog barking
Farm sounds
Cow mooing
Bird chirping
0.5
0
-0.5
Lobule VIIa crusI (Hem)
Lobule VIIa crusII (Hem)
Lobule VI (Hem)
Lobule VI (Verm)
Lobule VIIb (Hem)
Lobule VIIIa (Hem)
Lobule VIIIb (Hem)
Area FG1
Lobule VIIa crusII (Verm)
Lobule VIIIa (Verm)
Area FG2
Area hOc4v [V4(v)]
Area hOc3v [V3v]
Area hOc1 [V1]
Area hOc2 [V2]
Area hOc4la
Area hOc4lp
Area hOc4d [V3A]
Area hOc3d [V3d]
Area FG4
Area OP1 [SII]
Area OP4 [PV]
Area TE 1.0
Area TE 3
Area TE 1.2
Area TE 1.1
Area PF (IPL)
Area PFcm (IPL)
Area Ig2
Area PFm (IPL)
Area PGa (IPL)
Area 44
Area PFop (IPL)
Area PGp (IPL)
Area OP2 [PIVC]
Area 45
BF (Ch 4)
Area 1
7PC (SPL)
a 5L (SPL)
a 5M (SPL)
Area 4a
a 7A (SPL)
a 7P (SPL)
inal Cortex
Area s32
Area s24
Area Fo2
Area Fo3
Area Fo1
Area 33
Area Fp1
Area Fp2
0.9

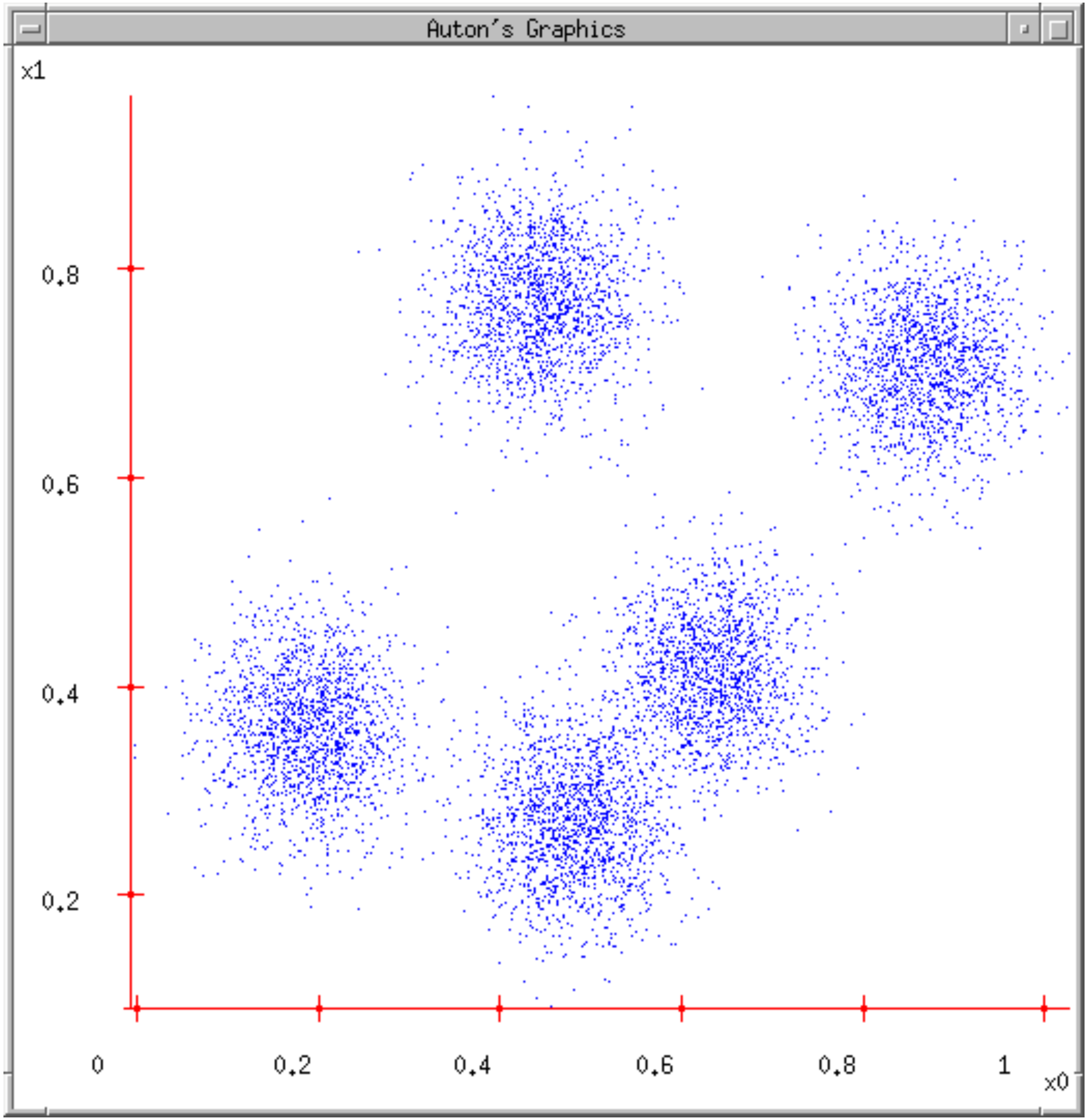


K-means Clustering

- Choose the number of clusters *k*
- Initialize cluster centers $\mu_1, \ldots \mu_k$
  - Randomly pick *k* data points and set cluster centers to these points
- For each data point, compute the cluster center it is closest to (using a distance measure) and assign the data point to this cluster
- Re-compute cluster centers (mean of data points in cluster)
- Stop when there are no new re-assignments

K-means

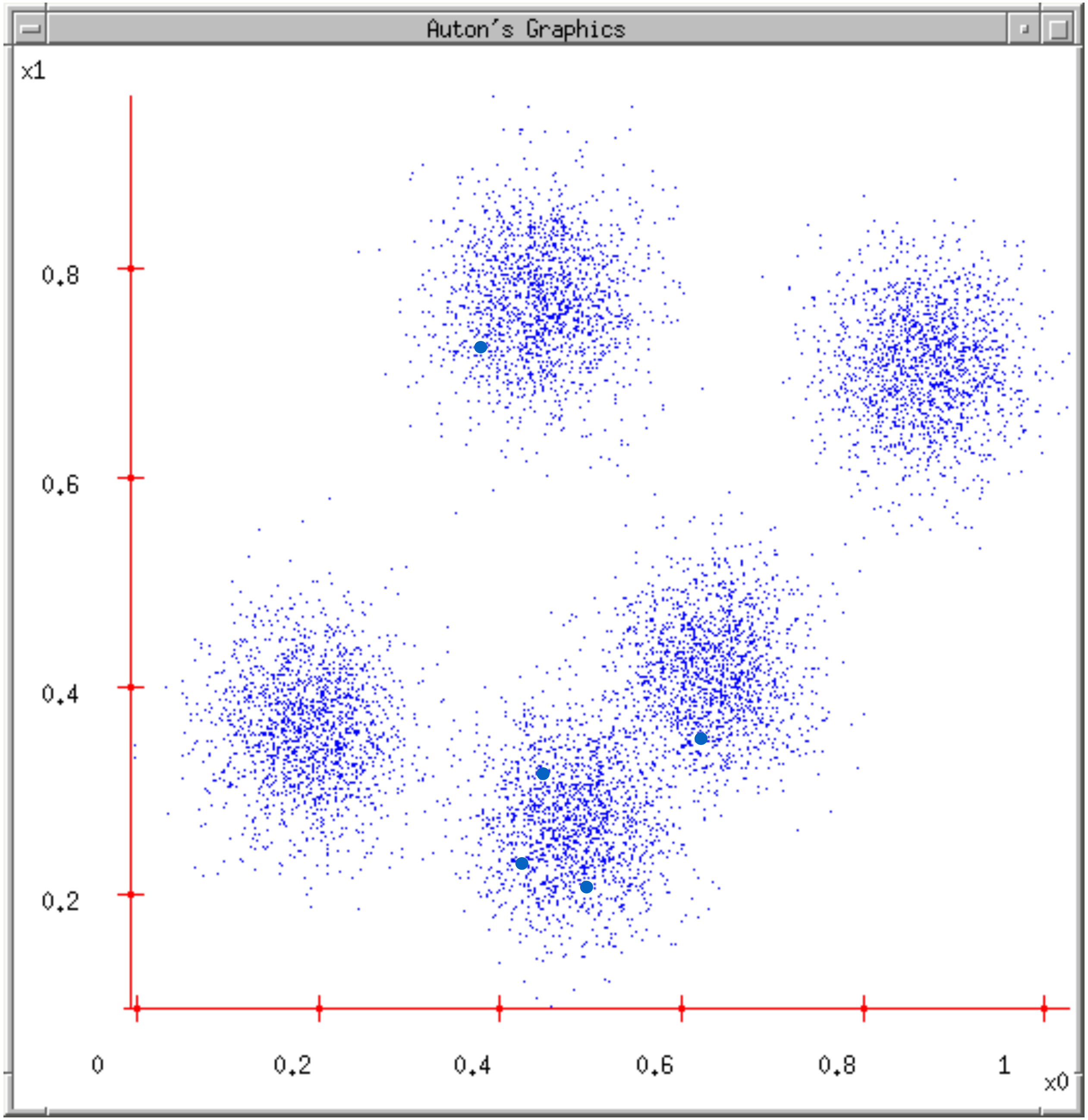
1. Ask user how many clusters they'd like. *(e.g. k=5)*



https://reurl.cc/b779v

K-means

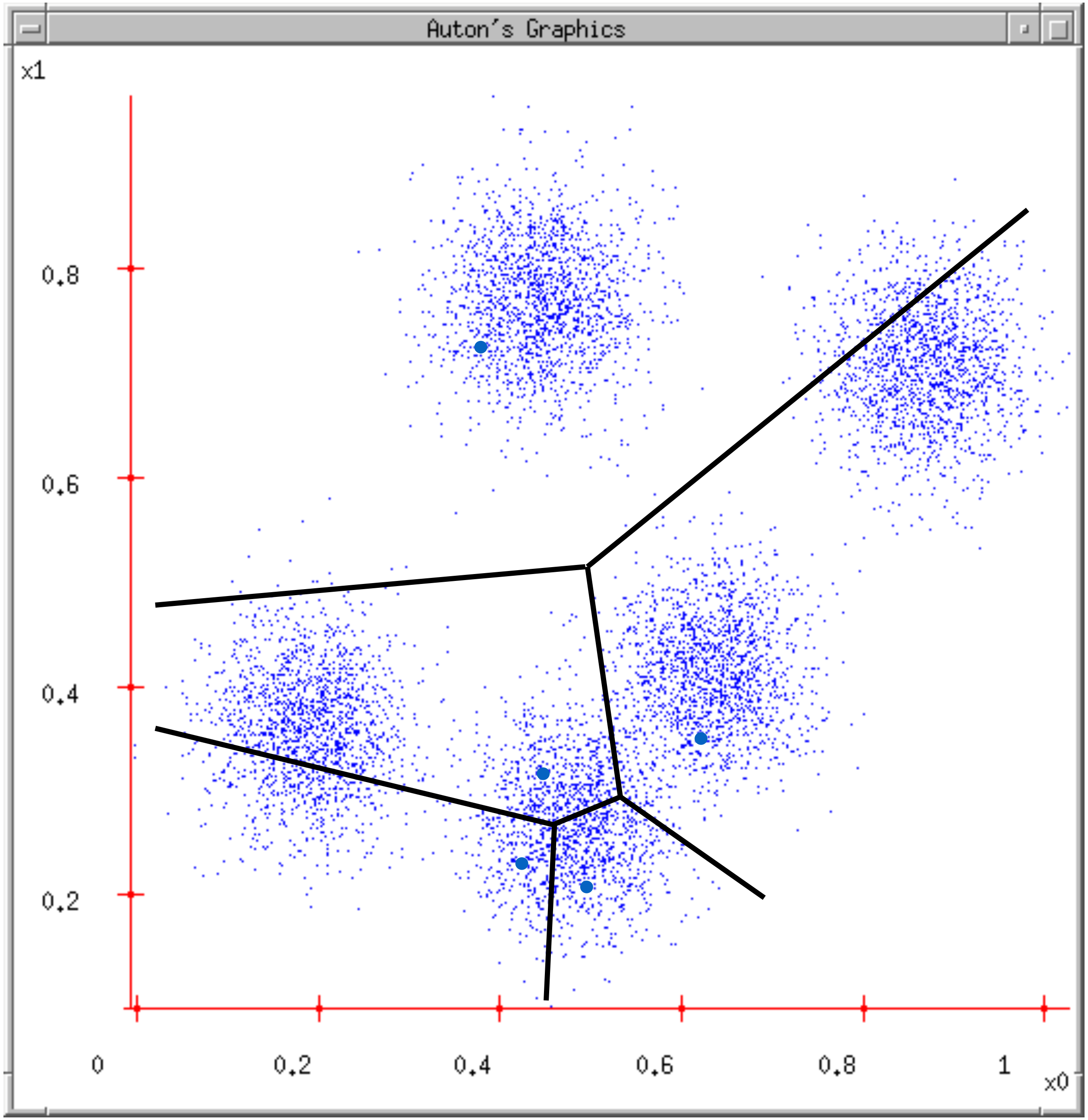
1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations



https://reurl.cc/b779v

K-means

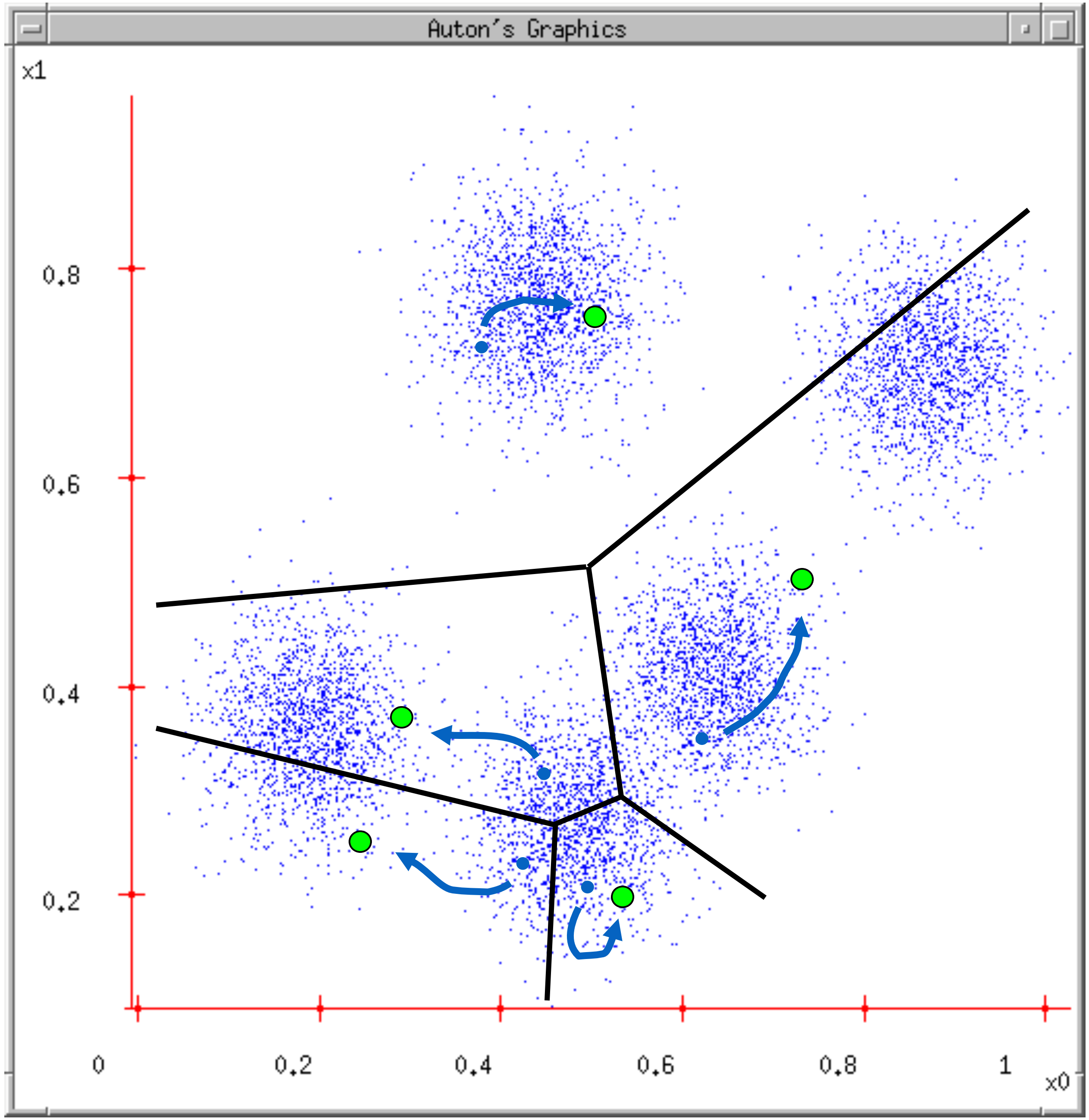
1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.



https://reurl.cc/b779v

K-means

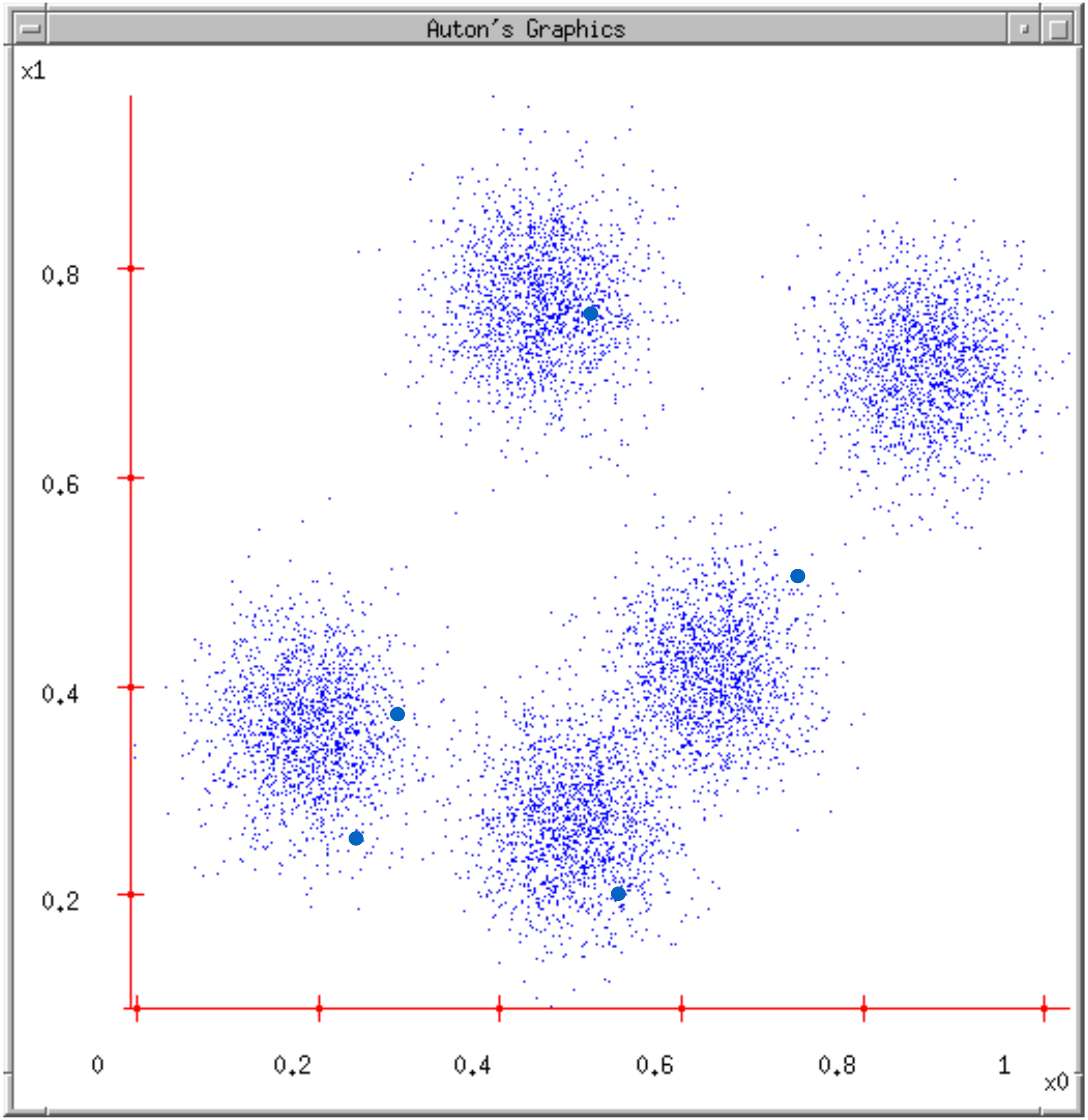
1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns



https://reurl.cc/b779v

K-means

1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns…
5. …and jumps there
6. …Repeat until terminated!

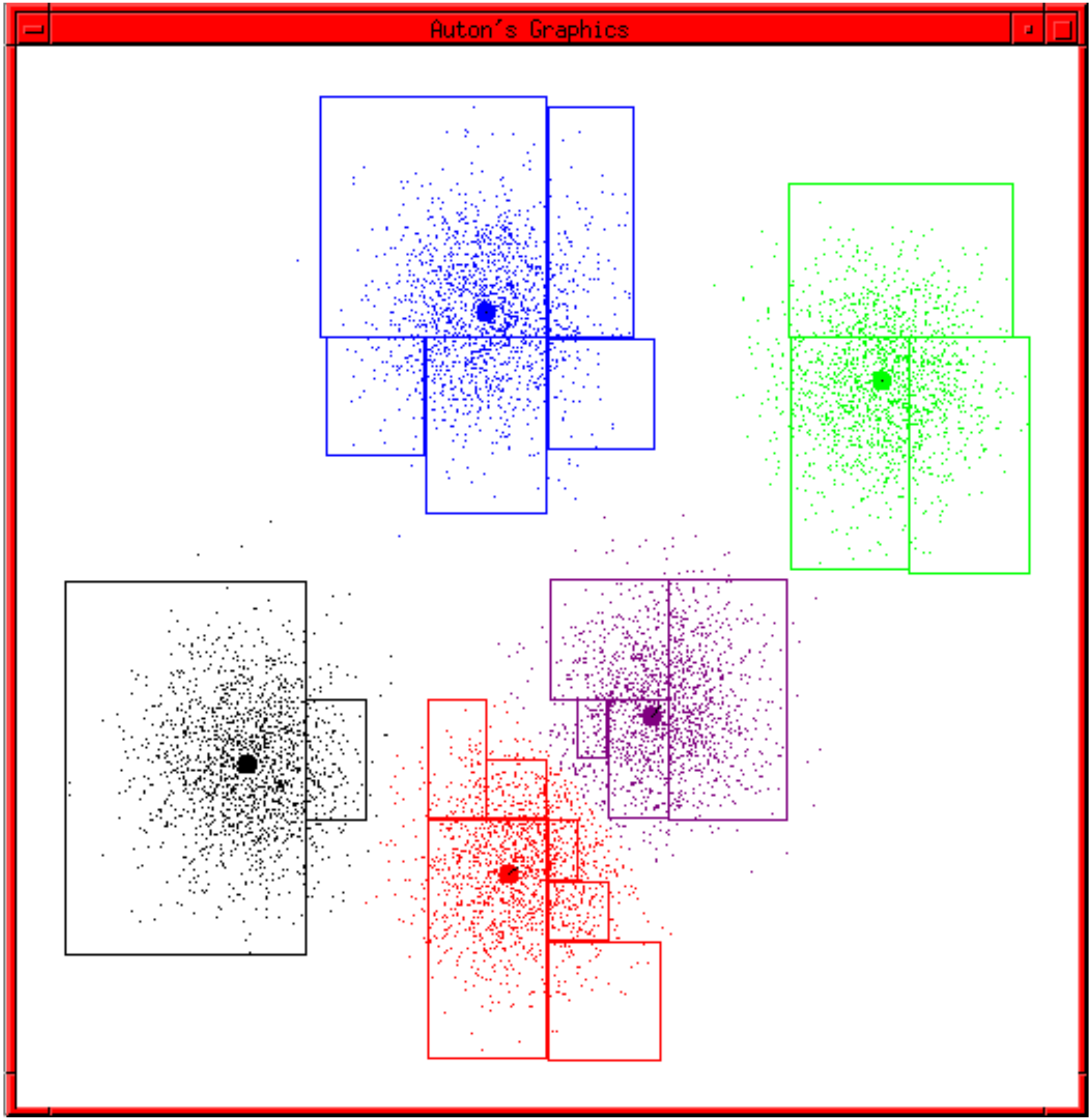


https://reurl.cc/b779v

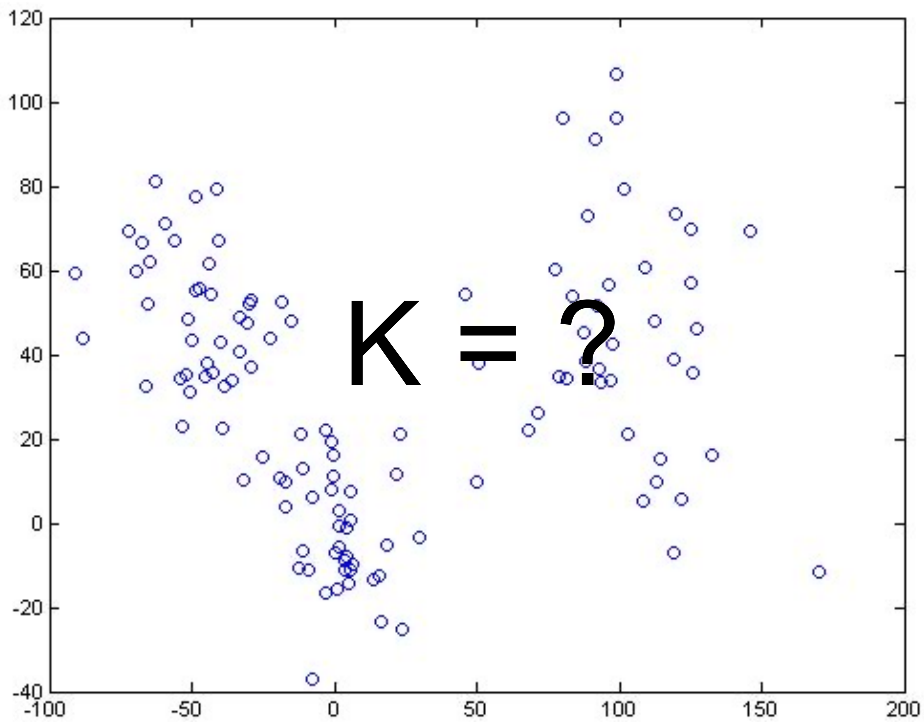
K-means

Example generated by Dan Pelleg's super-duper fast K-means system:

Dan Pelleg and Andrew Moore. Accelerating Exact k-means Algorithms with Geometric Reasoning. Proc. Conference on Knowledge Discovery in Databases 1999, (KDD99) (available on www.autonlab.org/pap.html*)*



K-means Clustering Issues



K = ?

How many clusters do you think there are in this data? How might it have been generated?

$$K \approx \sqrt{n/2}$$

K-means Clustering Issues

- Random initialization means that you may get different clusters each time
- Data points are assigned to only one cluster
- Implicit assumptions about the “shapes” of clusters
- You must pick the number of clusters…
- Will K-means always converge?

Determining K

- We'd like to have a measure of cluster quality Q and then try different values of k until we get an optimal value for Q
- This is an unsupervised learning method; we can't really find a "correct" measure Q…

Cluster Quality Measures

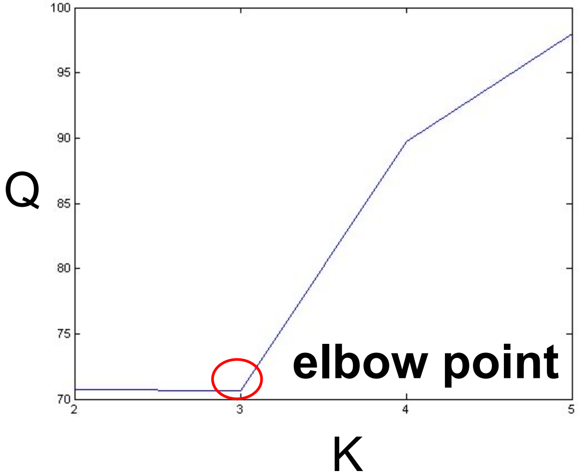
- A measure that emphasizes cluster tightness or homogeneity:

$$Q = \sum_{i=1}^{k} \frac{1}{|C_i|} \sum_{\boldsymbol{x} \in C_i} d(\boldsymbol{x}, \mu_i)$$

Similar to Ward's Method!

- $|C_i|$ is the number of data points in cluster *i*
- *Q* will be small if the data points in each cluster are close

- An alternate approach is to look at cluster **stability:**
    - Add random noise to the data many times and count how many pairs of data points no longer cluster together



Summary

- Clustering is a very popular method of microarray analysis and also a well-established statistical technique.
- Many variations on *k*-means, including algorithms in which clusters can be split and merged or that allow for soft assignments
- *Semi-supervised* clustering methods, in which some examples are assigned by hand to clusters.

Questions?

Ref: https://colab.research.google.com/github/jakevdp/PythonDataScienceHandbook/blob/master/notebooks/05.11-K-Means.ipynb