

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

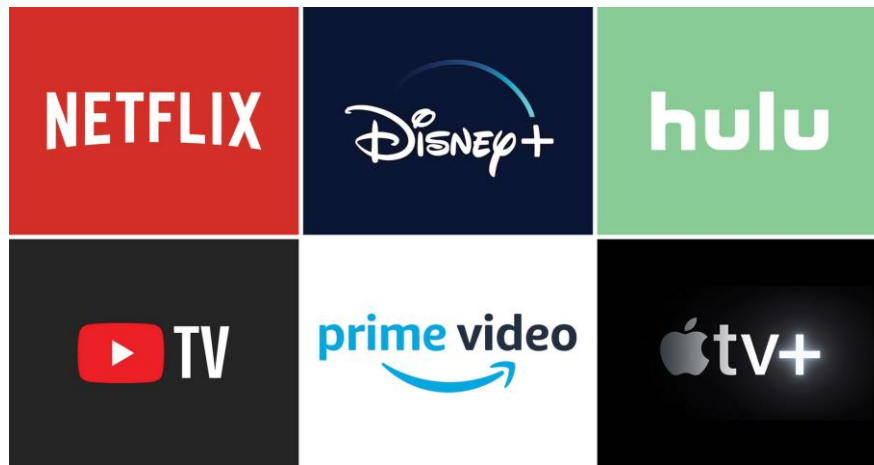
- Motivation
- Choosing (dis)similarity measures – **a critical step in clustering**
- Clustering algorithms

What is clustering?

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

Applications

- Streaming Services
 - To identify viewers who have similar behavior.



Minutes
watched
per day

Total viewing
sessions per
week

Number of unique
shows viewed per
month

Some applications

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

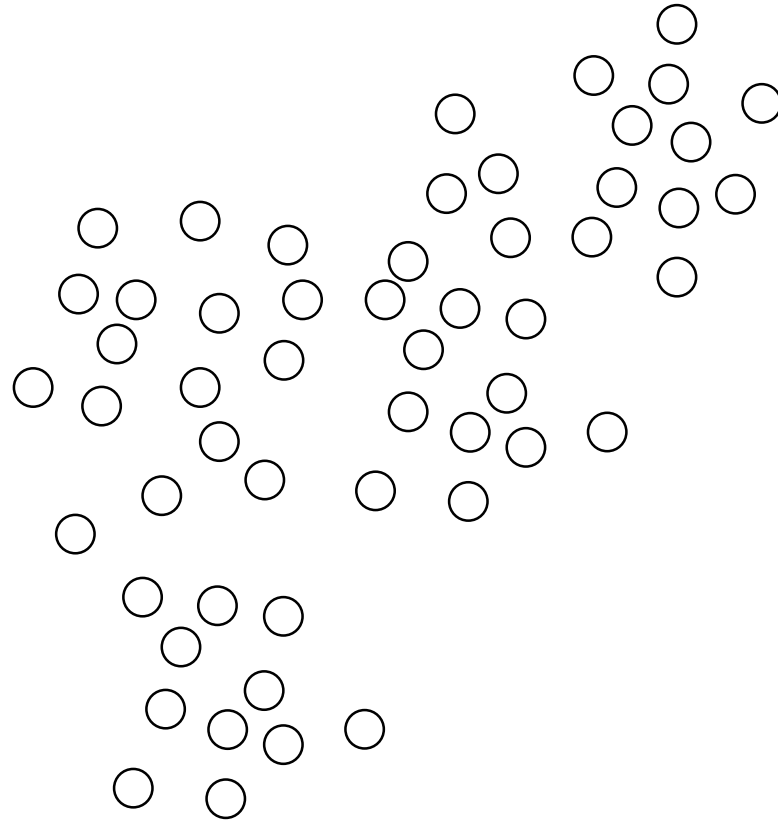
SEASON LEADERS						See All Player Stats
POINTS PER GAME		REBOUNDS PER GAME		ASSISTS PER GAME		
1. Joel Embiid PHI	32.0	1. Nikola Jokic DEN	12.8	1. Tyrese Haliburton IND	11.8	
2. Luka Doncic DAL	31.4	2. Anthony Davis LAL	12.5	2. Trae Young ATL	10.7	
3. Kevin Durant PHX	31.0	3. Domantas Sabonis SAC	11.8	3. Nikola Jokic DEN	9.8	
4. De'Aaron Fox SAC	30.3	4. Rudy Gobert MIN	11.6	4. Fred VanVleet HOU	9.1	
5. Giannis Antetokounmpo MIL	29.9	5. Joel Embiid PHI	11.3	5. Luka Doncic DAL	8.4	
BLOCKS PER GAME		STEALS PER GAME		FIELD GOAL PERCENTAGE		
1. Anthony Davis LAL	2.8	1. Shai Gilgeous-Alexander OKC	2.4	1. Jakob Poeltl TOR	72.9	
2. Brook Lopez MIL	2.8	2. Donovan Mitchell CLE	2.1	2. Daniel Gafford WAS	70.3	
3. Victor Wembanyama SAS	2.7	3. Jalen Suggs ORL	1.9	3. Jarrett Allen CLE	69.2	
4. Rudy Gobert MIN	2.4	4. Herbert Jones NOP	1.9	4. Mark Williams CHA	65.3	
5. Chet Holmgren OKC	2.2	5. Scottie Barnes TOR	1.8	5. Moritz Wagner ORL	63.3	
THREE POINTERS MADE		THREE POINT PERCENTAGE		FANTASY POINTS PER GAME		
1. Stephen Curry GSW	91	1. Cason Wallace OKC	52.5	Nikola Jokic DEN	62.1	
2. Luka Doncic DAL	69	2. Nicolas Batum PHI	51.3	Joel Embiid PHI	60.4	
3. Jalen Brunson NYK	63	3. Kevin Durant PHX	49.4	Luka Doncic DAL	55.8	
3. Tyrese Haliburton IND	63	4. Alex Caruso CHI	47.7	Giannis Antetokounmpo MIL	54.4	
5. Desmond Bane MEM	62	5. Doug McDermott SAS	47.5	Shai Gilgeous-Alexander OKC	54.0	

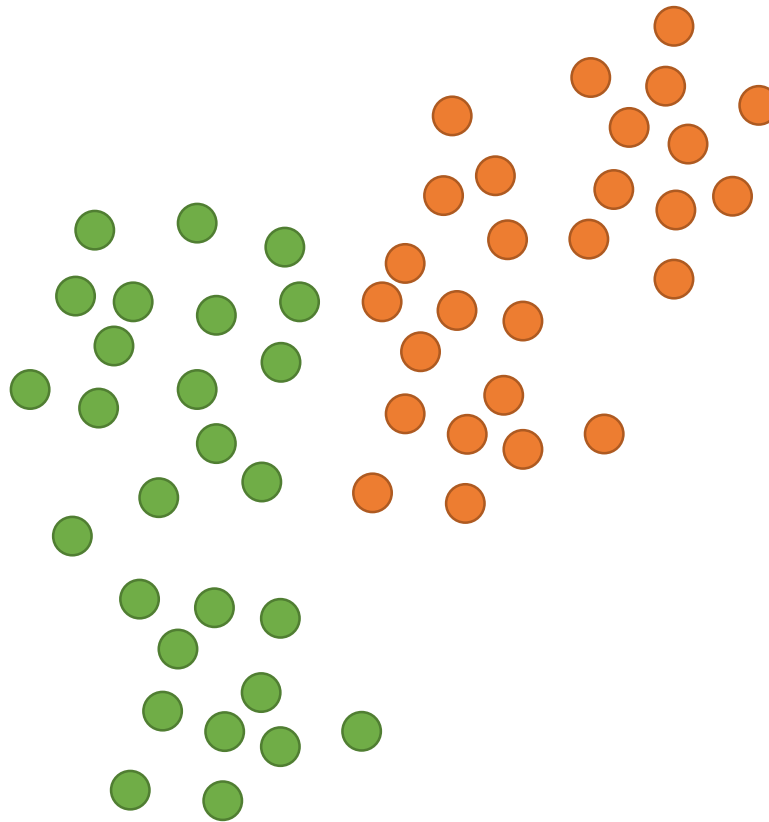
Group by features

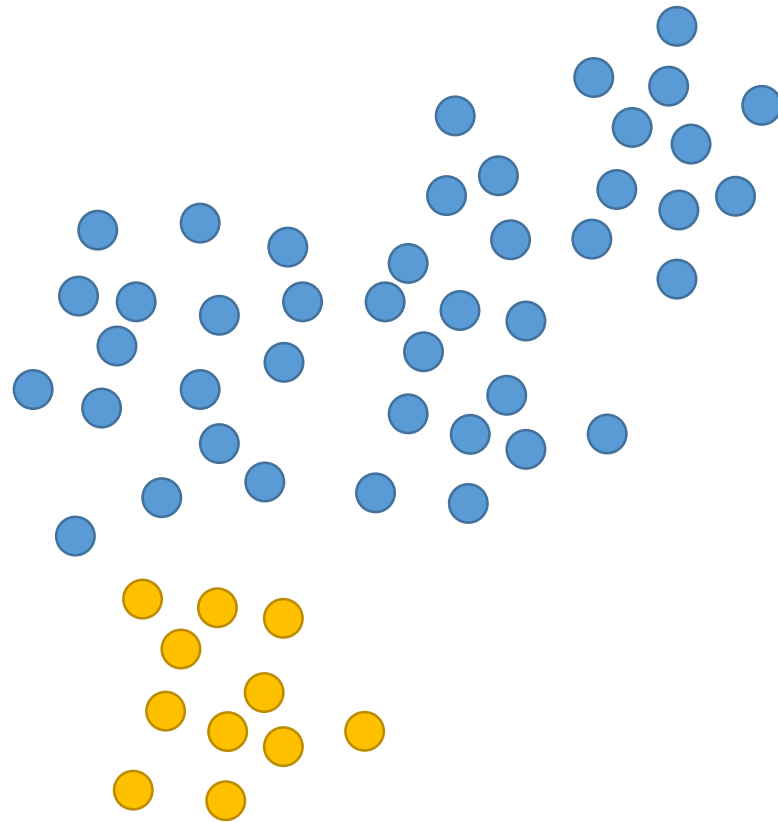
Example	Attributes										Target
	Alt.	Bar	Fri	Hun	Pat	Price	Rain	Res	Type	Est.	Wait
X ₁	T	F	F	T	Some	\$\$\$	F	T	French	0-10	
X ₂	T	F	F	T	Full	\$	F	F	Thai	30-60	
X ₃	F	T	F	F	Some	\$	F	F	Burger	0-10	
X ₄	T	F	T	T	Full	\$	F	F	Thai	10-30	
X ₅	T	F	T	F	Full	\$\$\$	F	T	French	>60	
X ₆	F	T	F	T	Some	\$\$	T	T	Italian	0-10	
X ₇	F	T	F	F	None	\$	T	F	Burger	0-10	
X ₈	F	F	F	T	Some	\$\$	T	T	Thai	0-10	
X ₉	F	T	T	F	Full	\$	T	F	Burger	>60	
X ₁₀	T	T	T	T	Full	\$\$\$	F	T	Italian	10-30	
X ₁₁	F	F	F	F	None	\$	F	F	Thai	0-10	
X ₁₂	T	T	T	T	Full	\$	F	F	Burger	30-60	

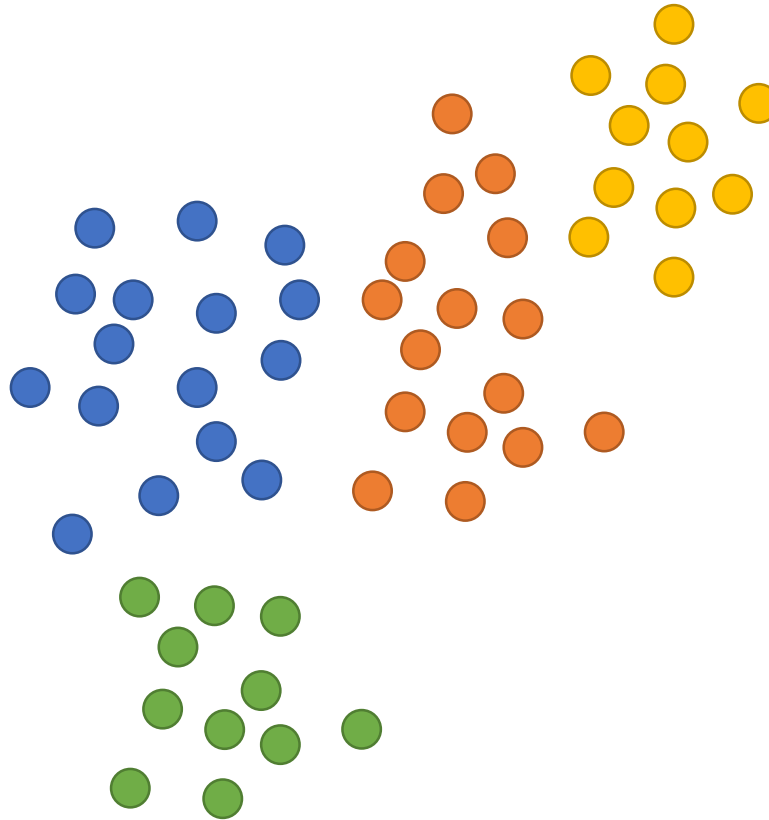
Group by instances

How to cluster the data?









There is no single right answer!

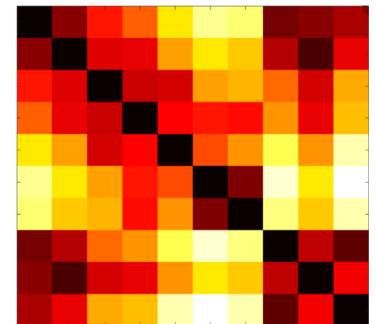
What degree of similarity is required for items to be placed in the same cluster?

Outline

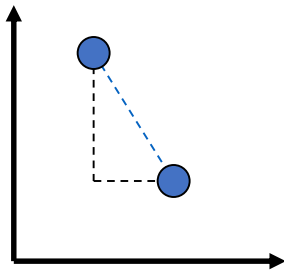
- Motivation
- Choosing (dis)similarity measures – **a critical step in clustering**
- Clustering algorithms

How do we define (dis)similarity?

- The goal is to group together “**similar**” data
- It depends on what we want to find or emphasize in the data;
- The similarity measure is often more important than the clustering algorithm used
- This is usually a ***pair-wise*** measure

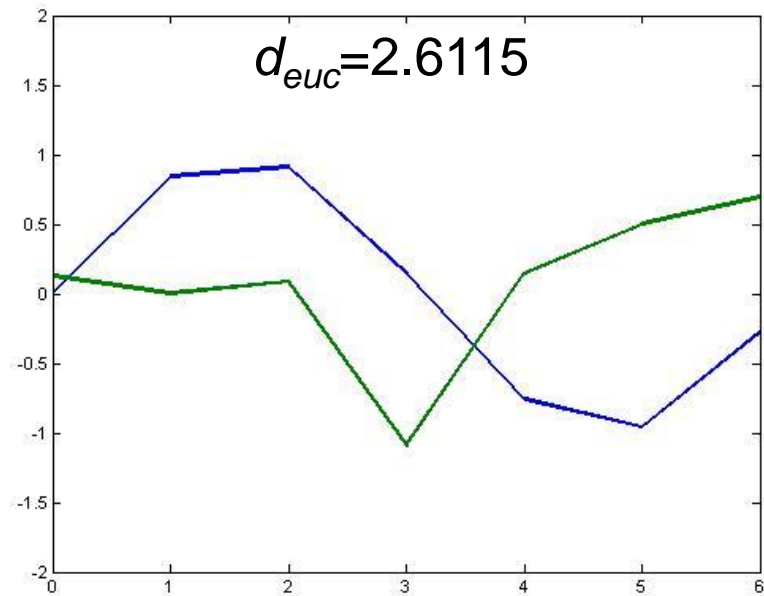
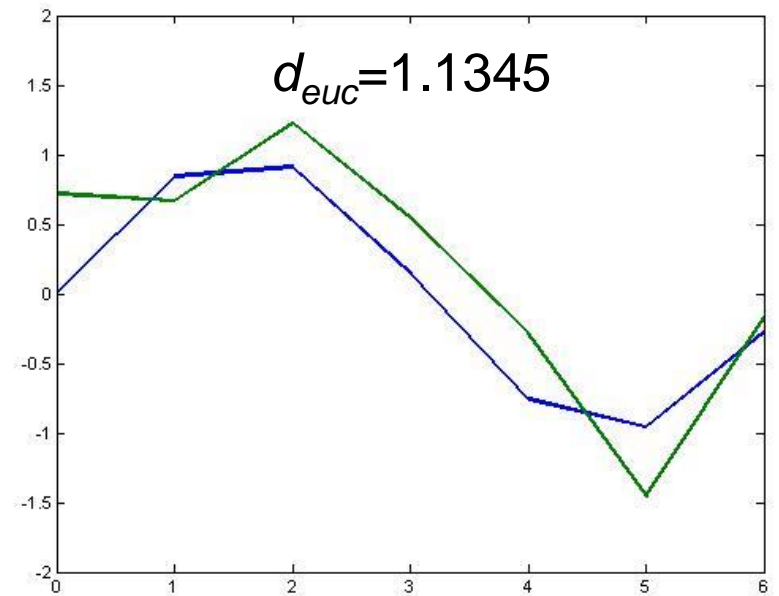
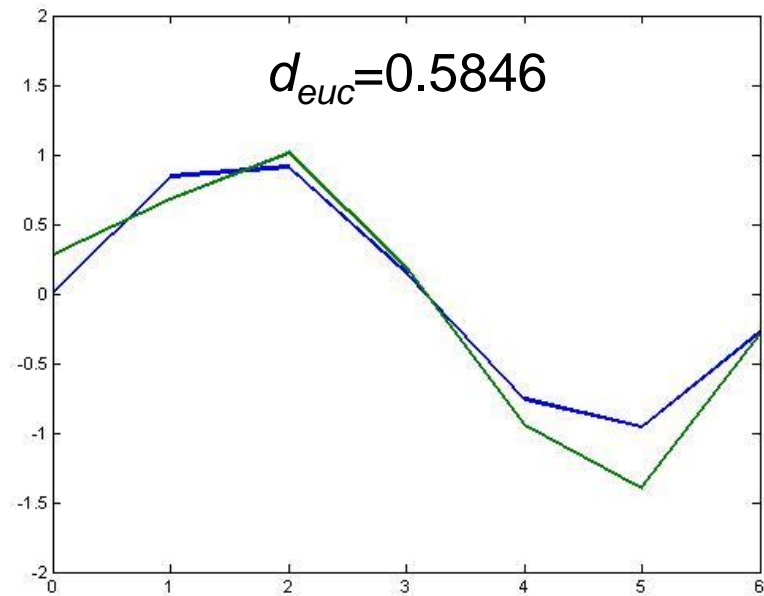


Euclidean distance



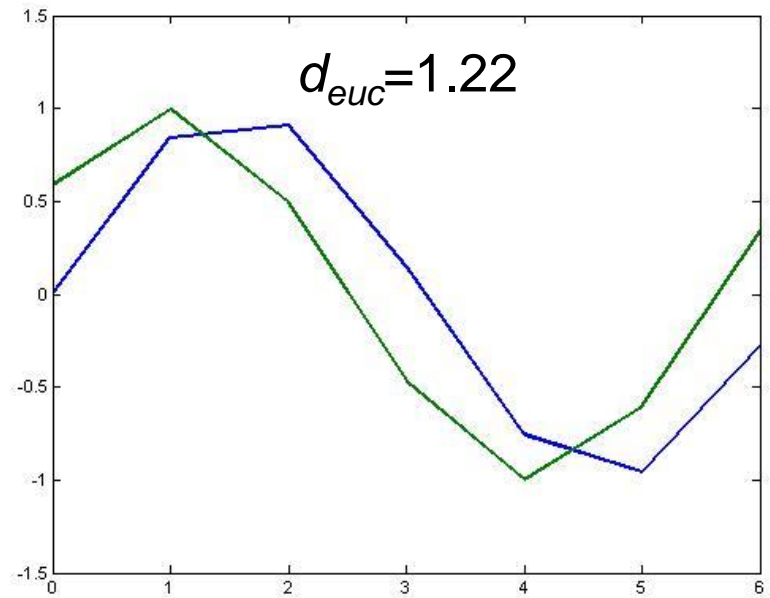
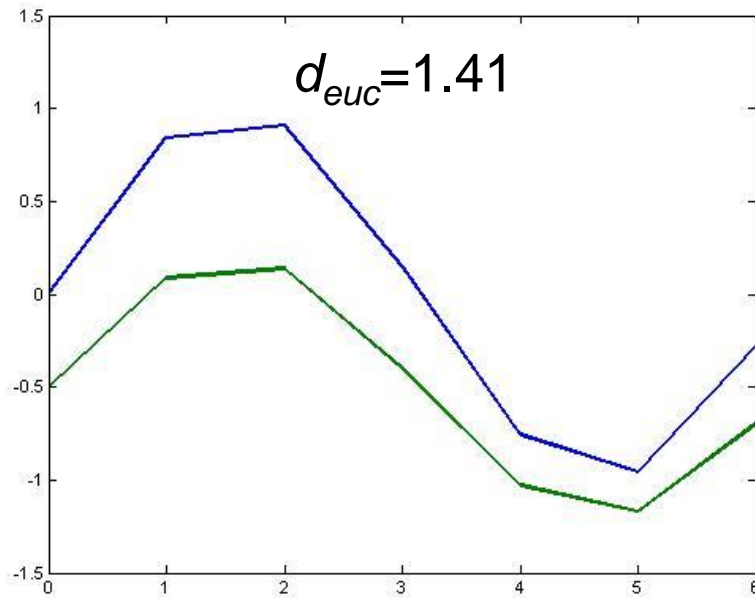
$$d_{euc}(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

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



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

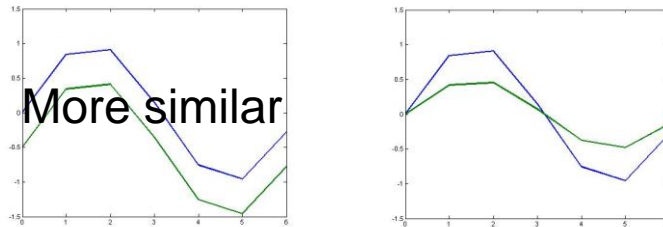
But what about these?



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

Pearson Correlation

- We might care more about the shape of data profiles rather than the magnitudes



- We can make the data have mean = 0 and std = 1

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

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

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

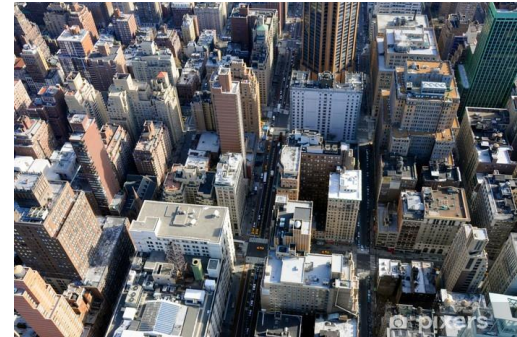
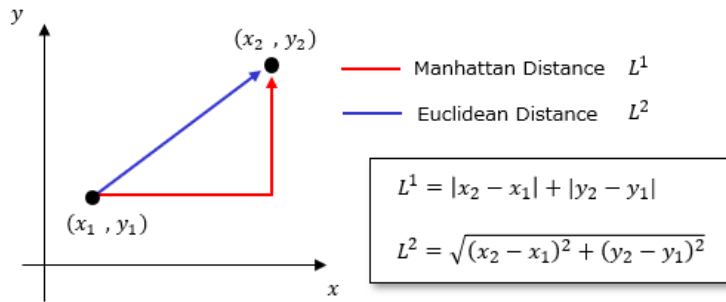
Pearson Correlation

- Pearson correlation is a measure that is invariant to scaling and shifting of the data values
- Always between -1 and $+1$
- We can easily make it into a dissimilarity measure:

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

Other measures

- Manhattan distance (or Cityblock, or L1), cosine distance



Manhattan is preferred over Euclidean distance:

1. High dimensional data.
2. Data points are not evenly distributed across all dimensions.

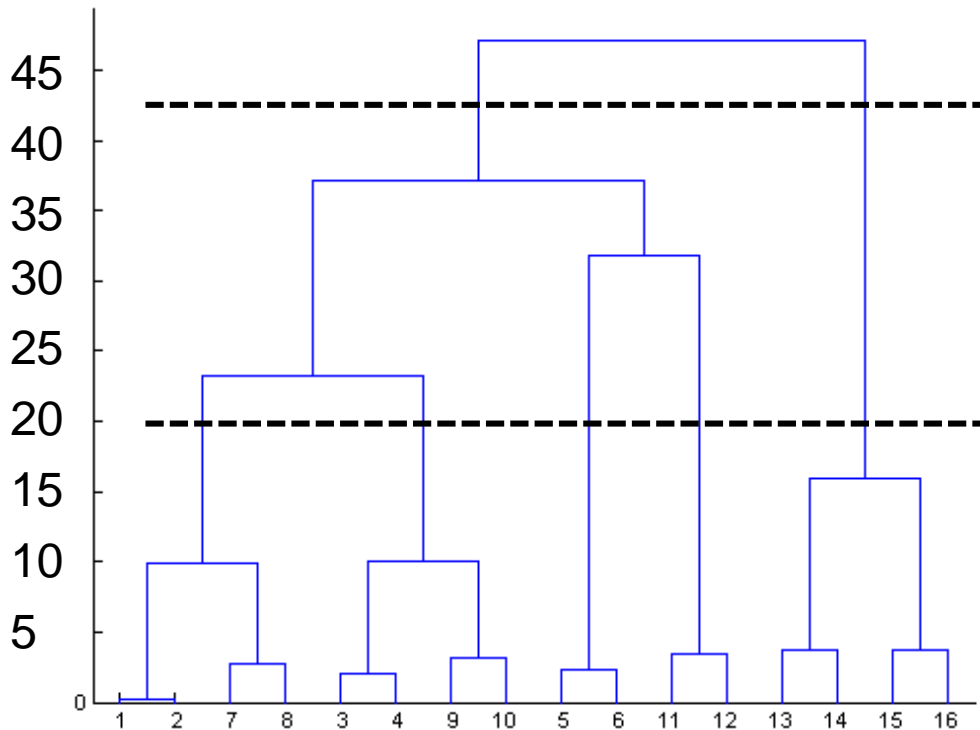
Outline

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

Hierarchical Clustering

- Start with every data point in a separate cluster
- 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.)



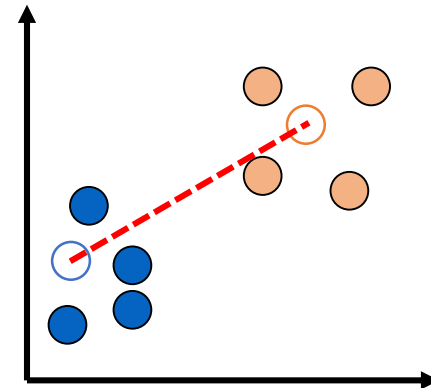
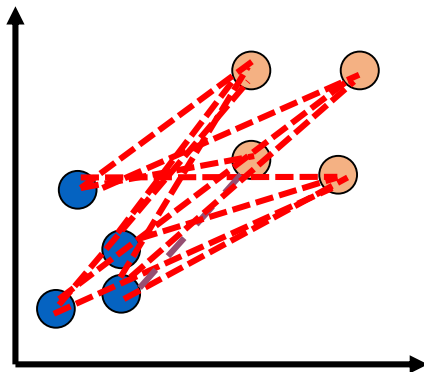
- This produces a binary tree or ***dendrogram***
- The final cluster is the root and each data item is a leaf
- The height of the bars indicate how close the items are

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

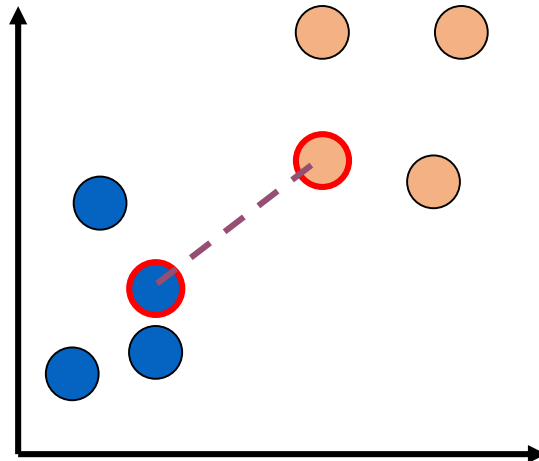
Average Linkage

- 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 $d(\mu_i, \mu_j)$



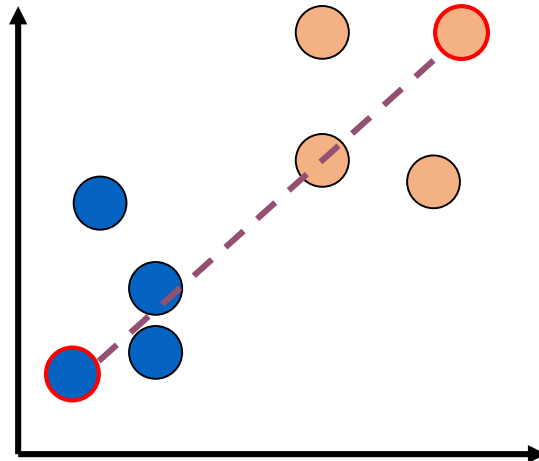
Single Linkage

- The **minimum** of all pairwise distances between points in the two clusters
- Tends to produce **loose** clusters



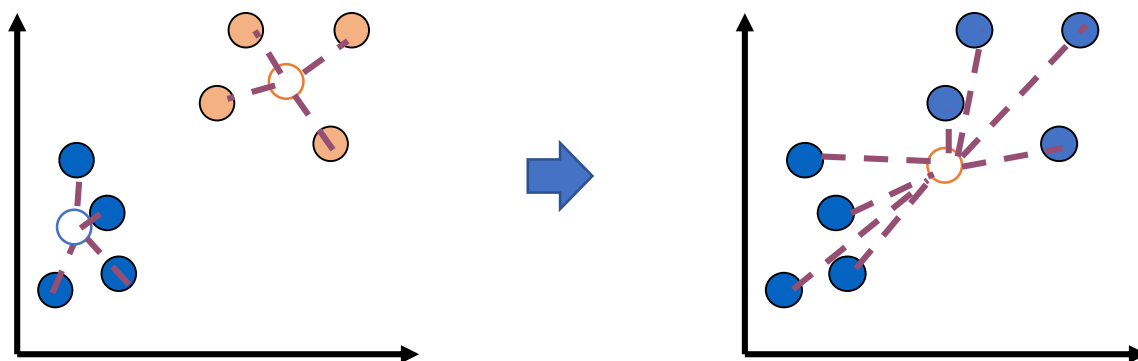
Complete Linkage

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

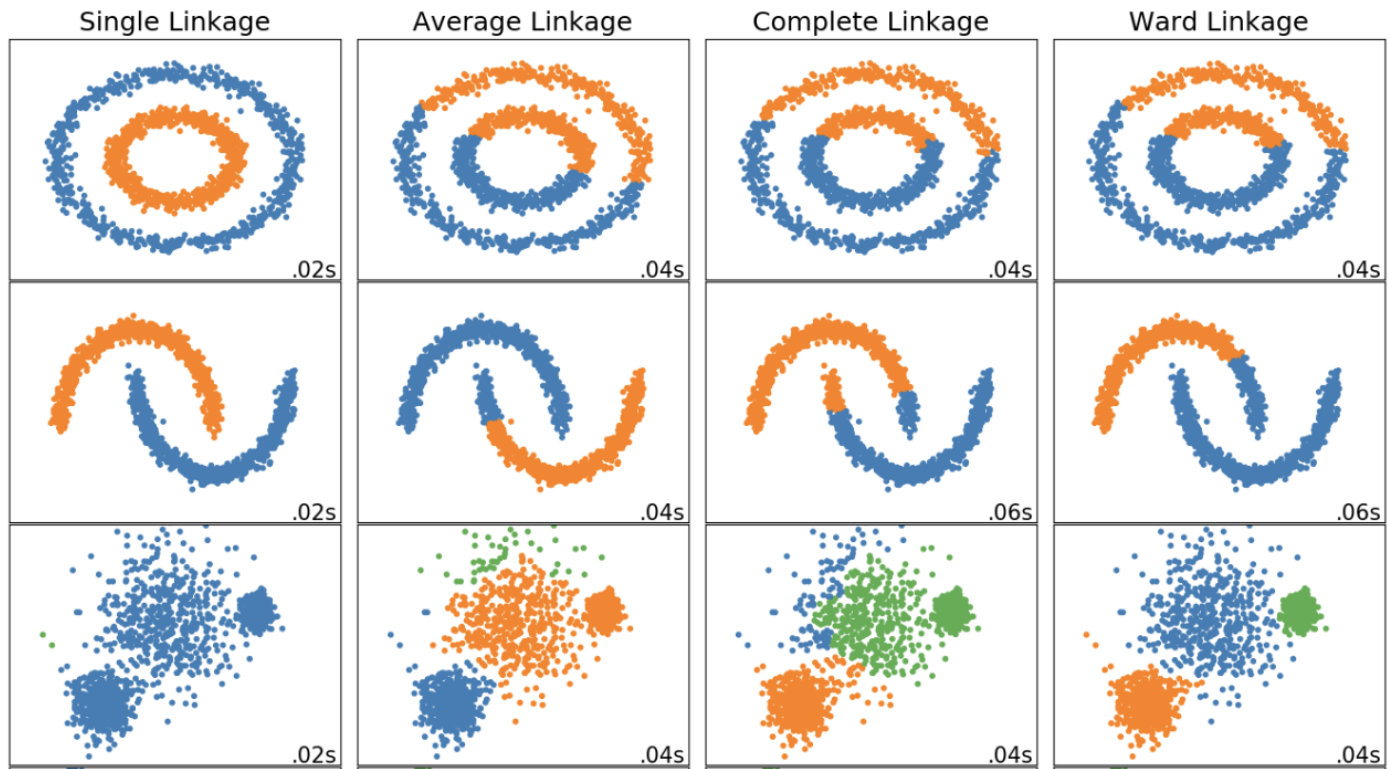


Ward's Method

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

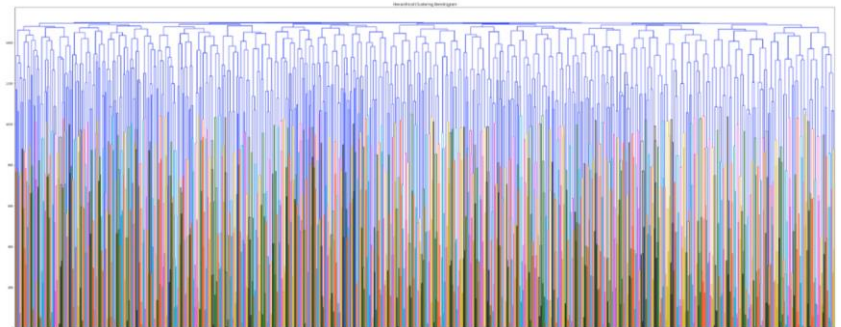


1. Find the centroid of each cluster.
2. Calculate the distance between each object and its cluster's centroid.
3. Calculate the sum of squared differences from Step 2.
4. Add up all the sums from Step 3.



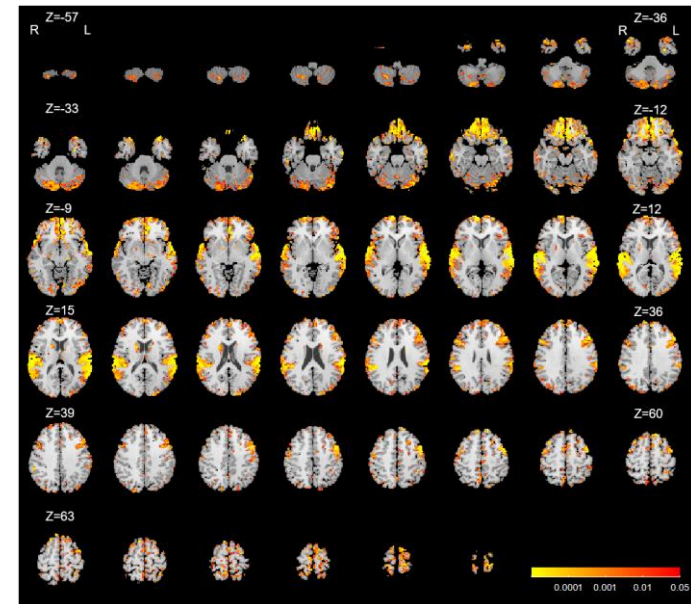
Hierarchical Clustering Issues

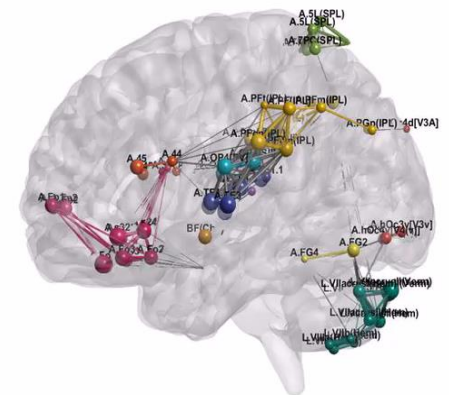
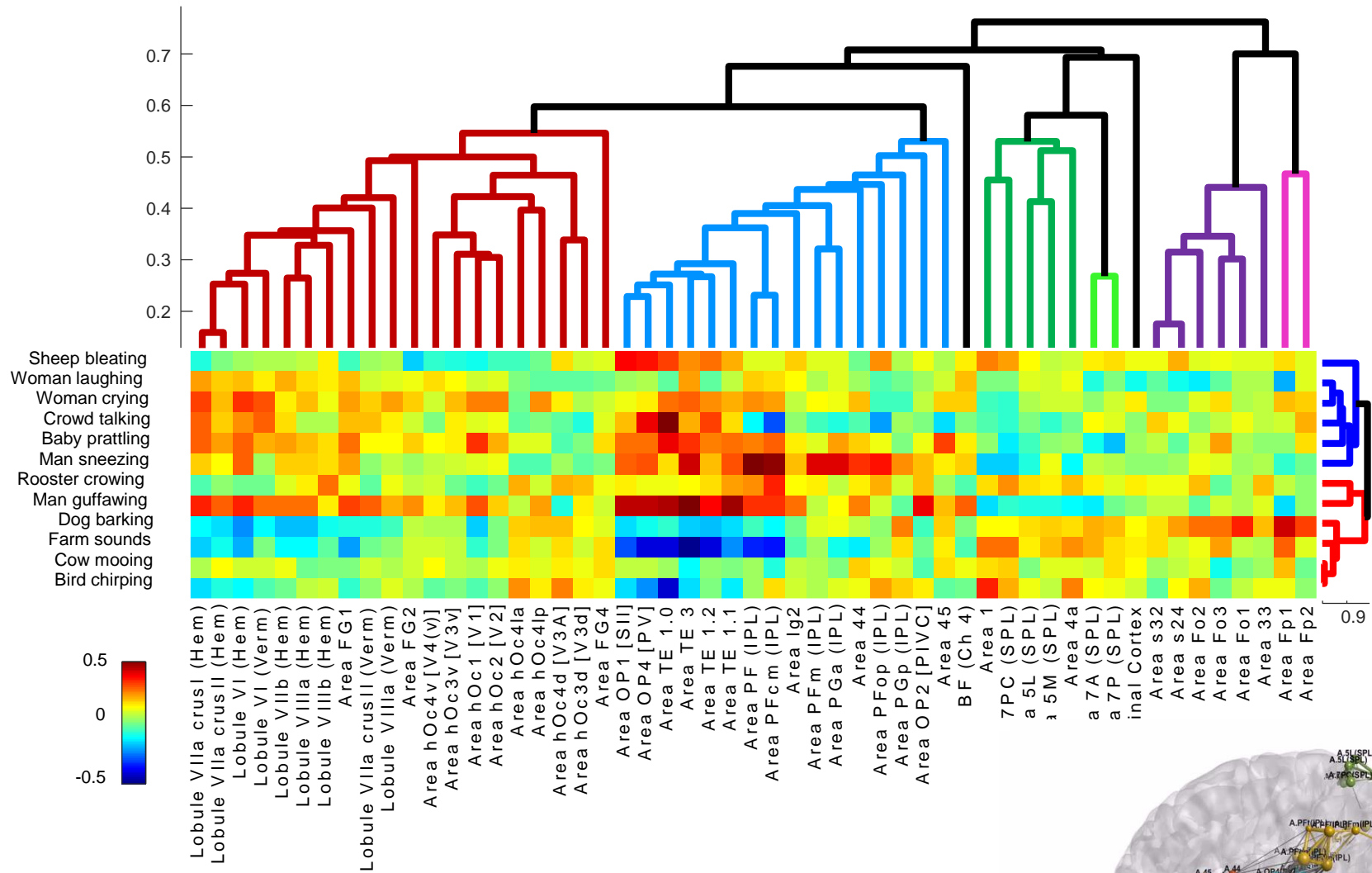
- Distinct clusters are not produced – sometimes this can be good, if the data has a hierarchical structure w/o clear boundaries (**No need to present the number of clusters**)
- There are methods for producing distinct clusters, but these usually involve specifying arbitrary **cutoff values**
- Heavy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



Example

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



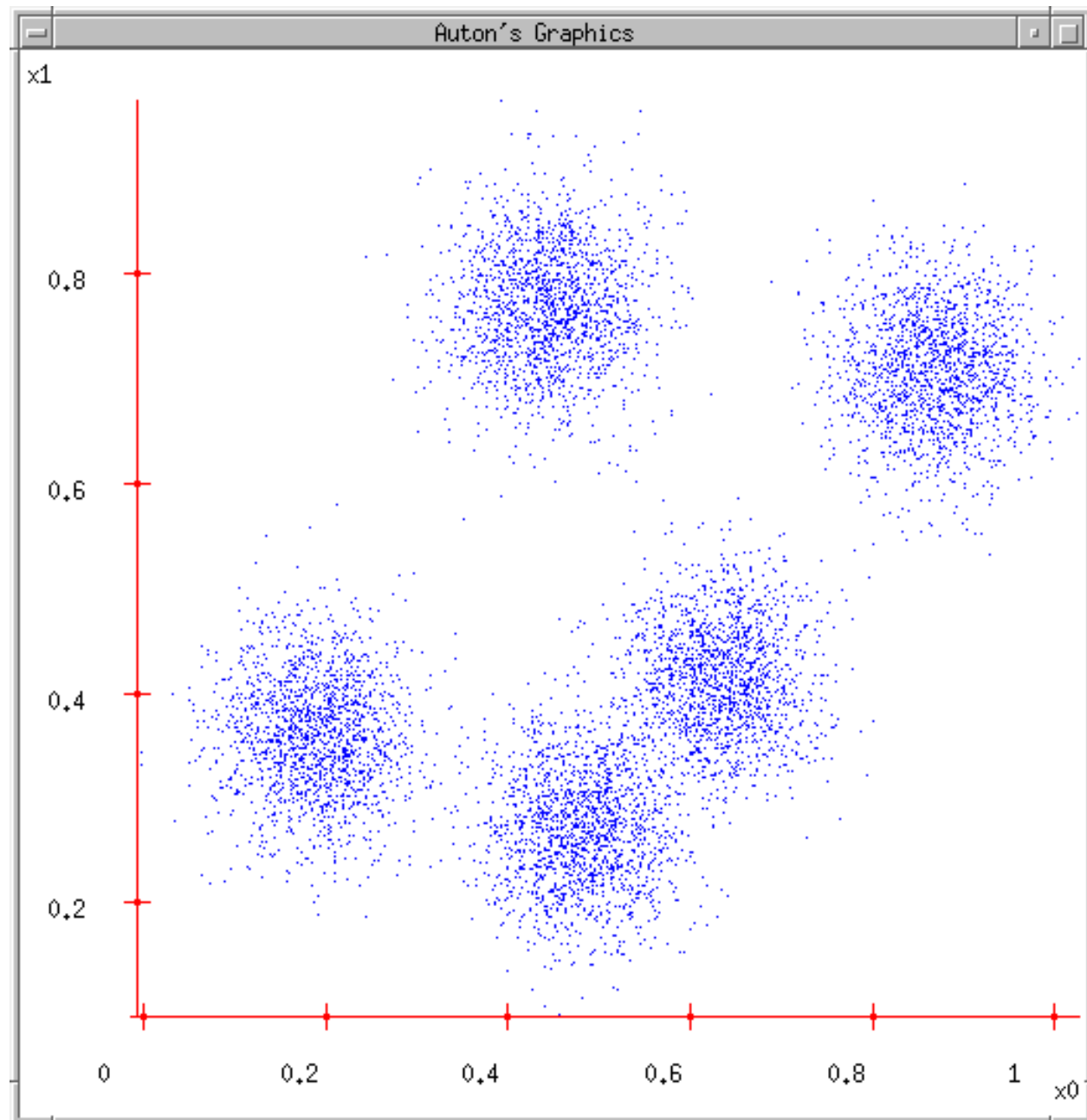


K-means Clustering

- Choose the number of clusters k
- Initialize cluster centers μ_1, \dots, μ_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

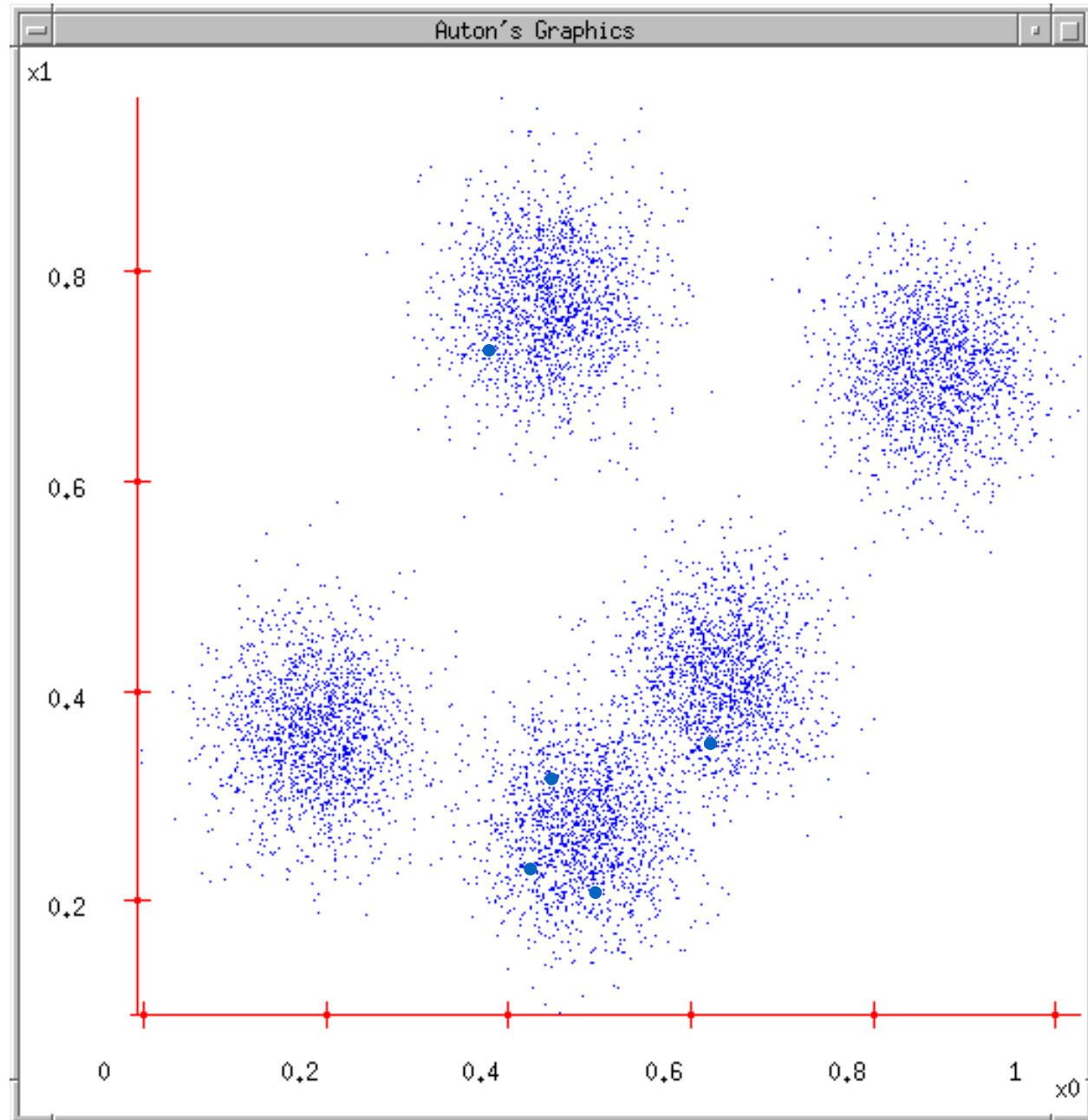
K-means

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



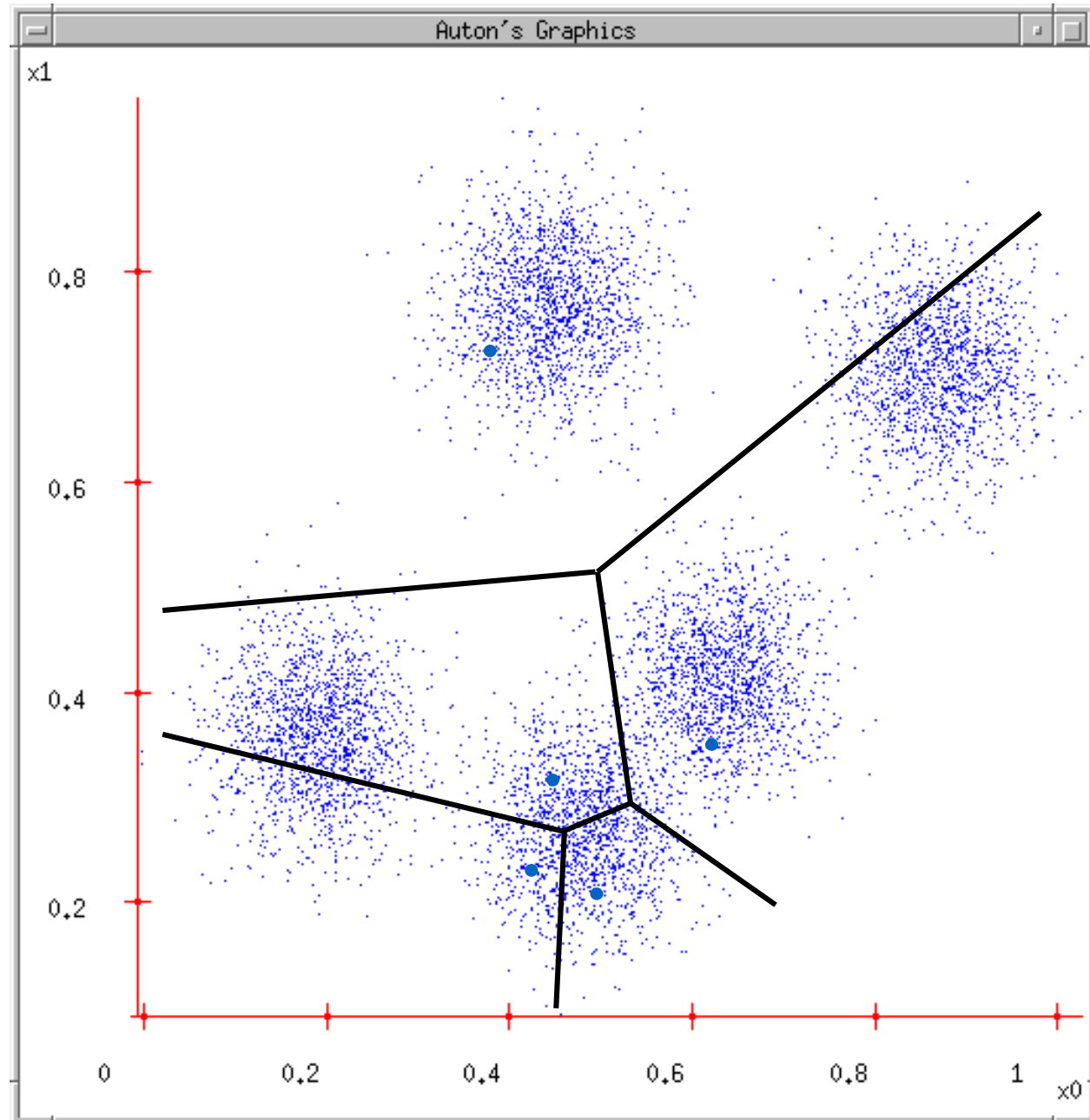
K-means

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



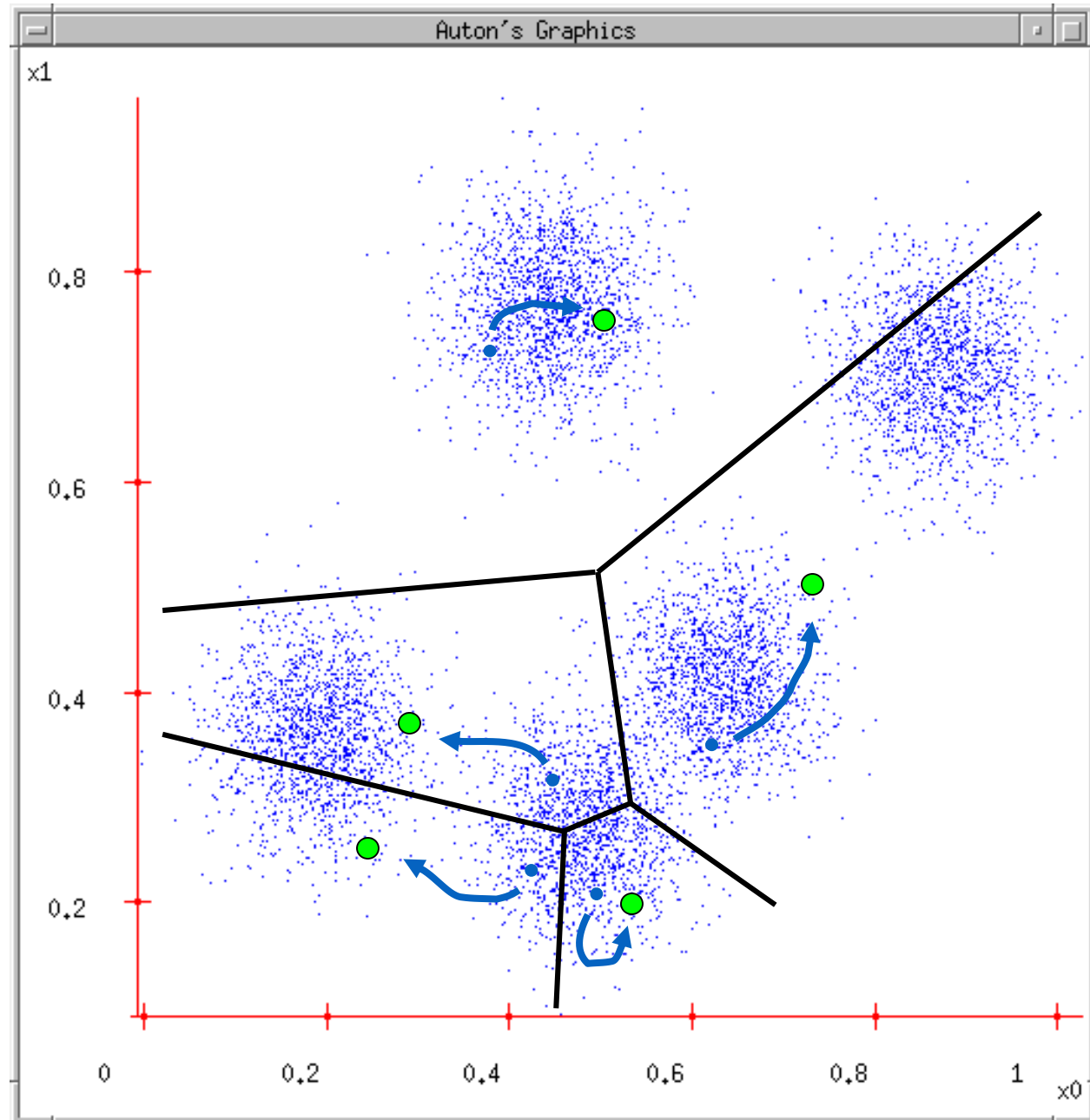
K-means

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.



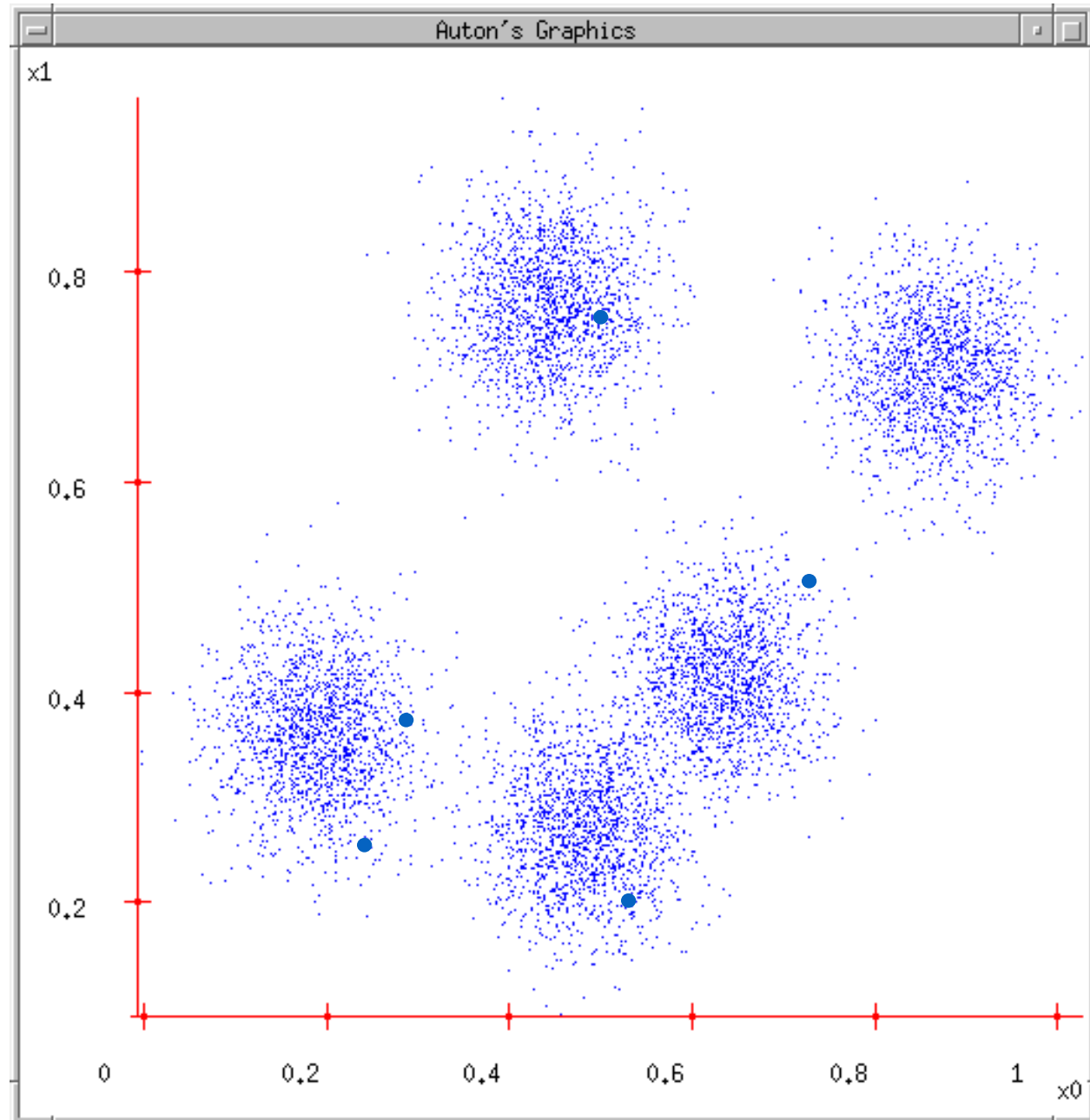
K-means

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



K-means

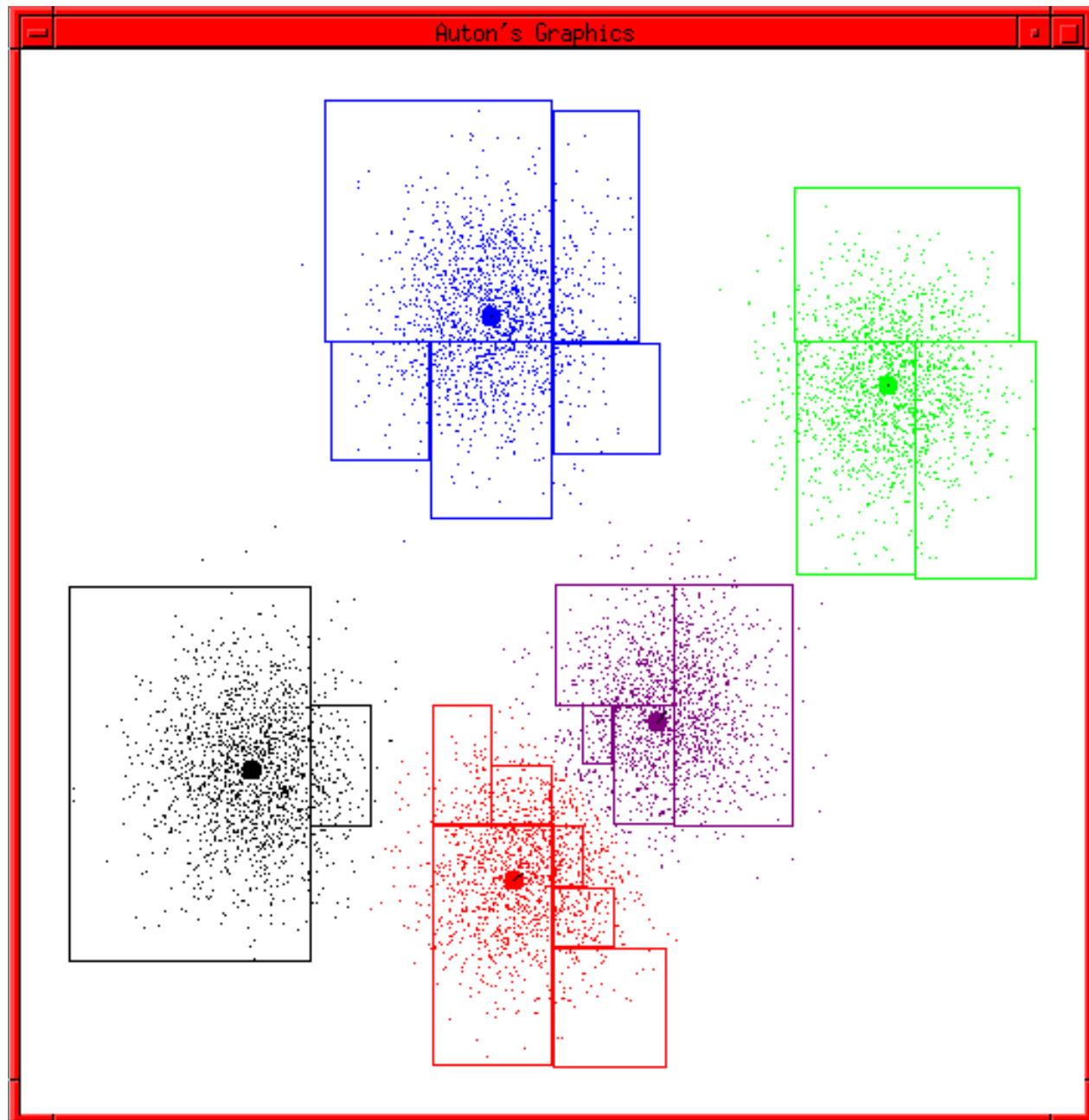
1. Ask user how many clusters they'd like.
(e.g. $k=5$)
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns...
5. ...and jumps there
6. ...Repeat until terminated!



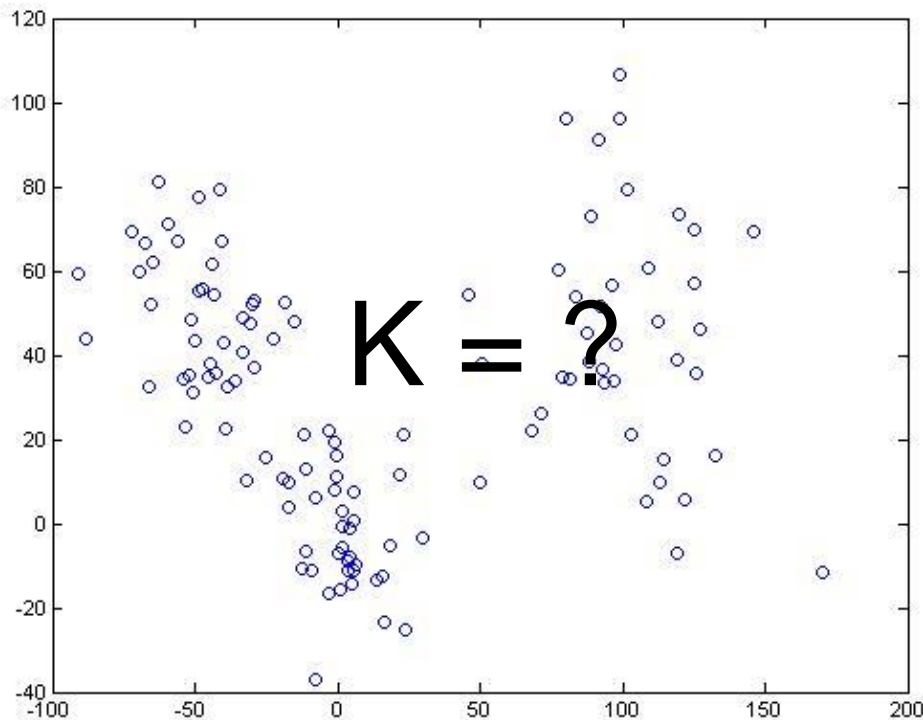
K-means

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



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



A cartoon character of a blue alien with a large head, wearing an orange shirt and blue pants, standing with hands on hips and looking up at the formula.

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

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

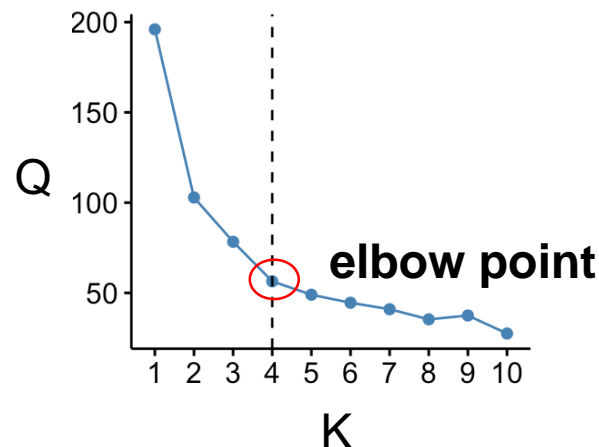
- A measure that emphasizes cluster tightness or homogeneity:

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

Similar to Ward's Method!

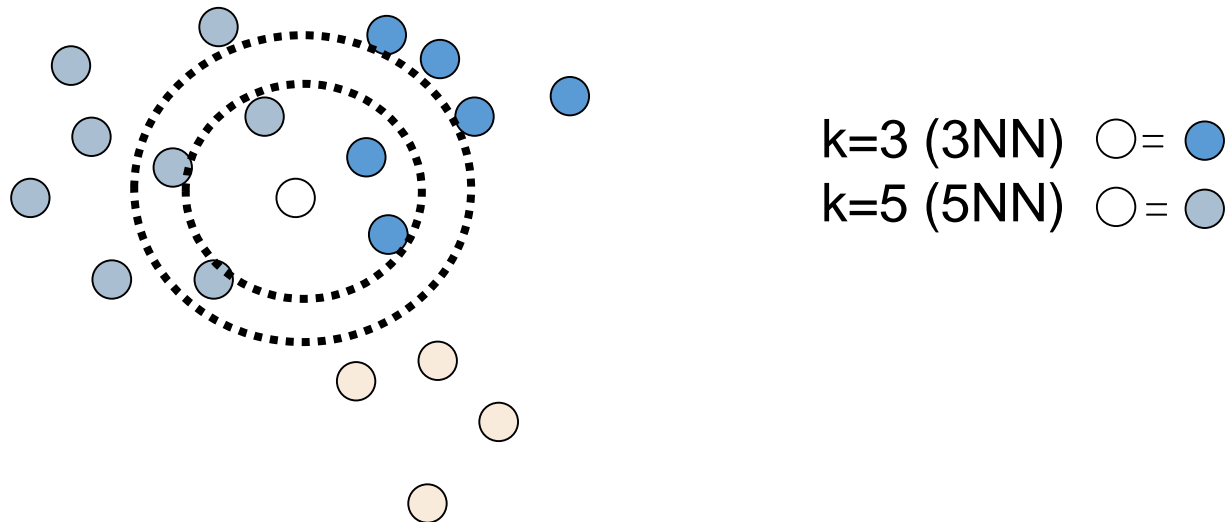


- $|C_i|$ is the number of data points in cluster i
- Q will be small if the data points in each cluster are close



k-Nearest Neighbor (kNN)

- A supervised learning classifier



Summary

- Clustering is a very popular method of biomedical (e.g., microarray) analysis.
- Many variations on *k*-means, including algorithms in which clusters can be split and merged.
- Clustering algorithm can be used for classification. How?

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

Questions?

Did you finish your prohject?

