Cluster Analysis

Potyó László

What is Cluster Analysis ?

- Cluster: a collection of data objects
  - Similar to one another within the same cluster
  - Dissimilar to the objects in other clusters
- Cluster analysis
  - Grouping a set of data objects into clusters
- Number of possible clusters (Bell) $O(e^{n \lg n})$
- Clustering is *unsupervised* classification: no predefined classes

General Applications of Clustering

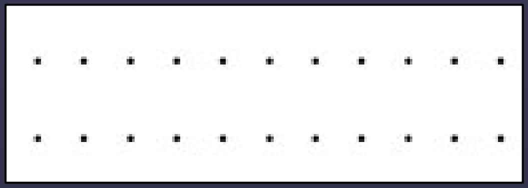
- Pattern Recognition
- Spatial Data Analysis
- Image Processing
- Economic Science
- WWW

Examples of Clustering Applications

- **Marketing:** Help marketers discover distinct groups in their customer bases, and then use this knowledge to develop targeted marketing program
- **Insurance:** Identifying groups of motor insurance policy holders with a high average claim cost
- **City-planning:** Identifying groups of houses according to their house type, value, and geographical location

What Is Good Clustering?

- The quality of a clustering result depends on both the similarity measure used by the method and its implementation.
- The quality of a clustering method is also measured by its ability to discover some or all of the hidden patterns.
- Example:



Requirements of Clustering

- Scalability
- Ability to deal with different types of attributes
- Discovery of clusters with arbitrary shape
- Minimal requirements for domain knowledge to determine input parameters
- Able to deal with noise and outliers
- Insensitive to order of input records
- High dimensionality
- Incorporation of user-specified constraints
- Interpretability and usability

Similarity and Dissimilarity Between Objects

- *Distances* are normally used to measure the similarity or dissimilarity between two data objects
- Some popular ones include:
  - *Minkowski distance*:

$$d(i,j) = \sqrt[q]{(|x_{i_1} - x_{j_1}|^q + |x_{i_2} - x_{j_2}|^q + ... + |x_{i_p} - x_{j_p}|^q)}$$

where $i = (x_{i1}, x_{i2}, \ldots, x_{ip})$ and $j = (x_{j1}, x_{j2}, \ldots, x_{jp})$ are two p-dimensional data objects, and q is a positive integer

Similarity and Dissimilarity Between Objects

- If *q* = *1*, *d* is Manhattan distance

$$d(i,j) = |x_{i_1} - x_{j_1}| + |x_{i_2} - x_{j_2}| + ... + |x_{i_p} - x_{j_p}|$$

- If *q* = *2*, *d* is Euclidean distance

$$d(i,j) = \sqrt{(|x_{i_1} - x_{j_1}|^2 + |x_{i_2} - x_{j_2}|^2 + ... + |x_{i_p} - x_{j_p}|^2)}$$

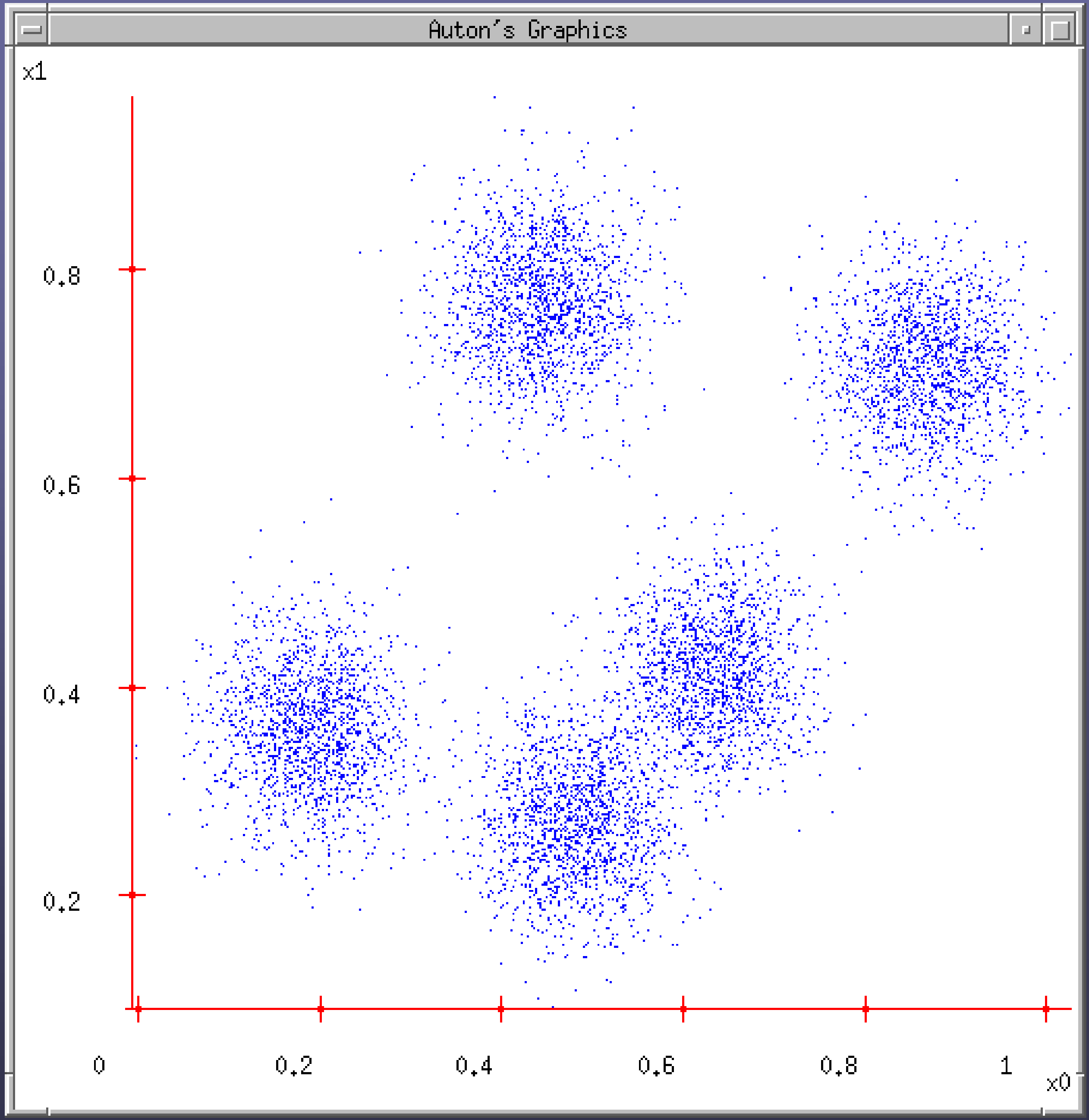
  - $d(i,j) \geq 0$
  - $d(i,i) = 0$
  - $d(i,j) = d(j,i)$
  - $d(i,j) \leq d(i,k) + d(k,j)$

Categorization of Clustering Methods

- Partitioning Methods
- Hierarchical Methods
- Density-Based Methods
- Grid-Based Methods
- Model-Based Clustering Methods

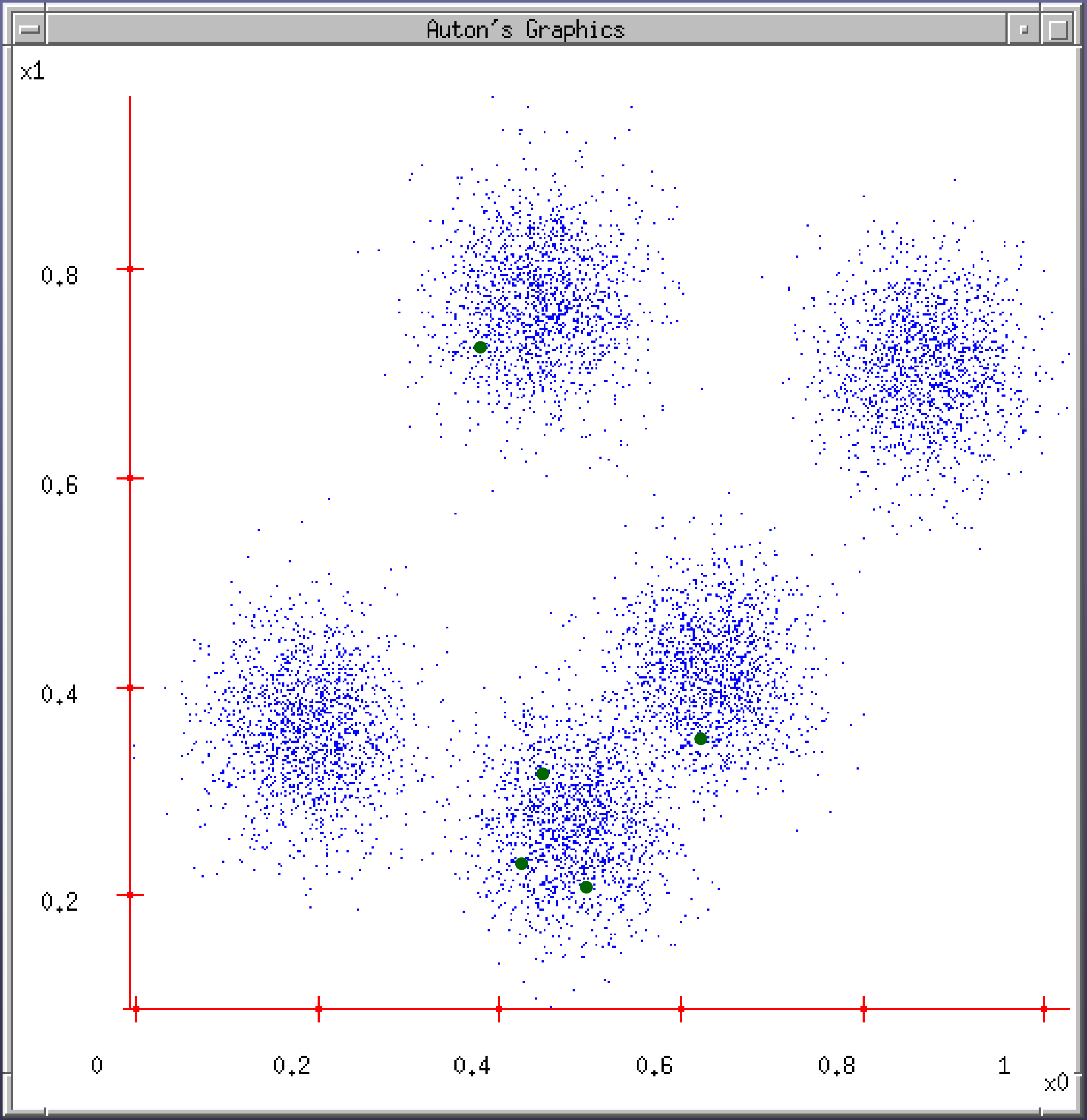
K-means

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



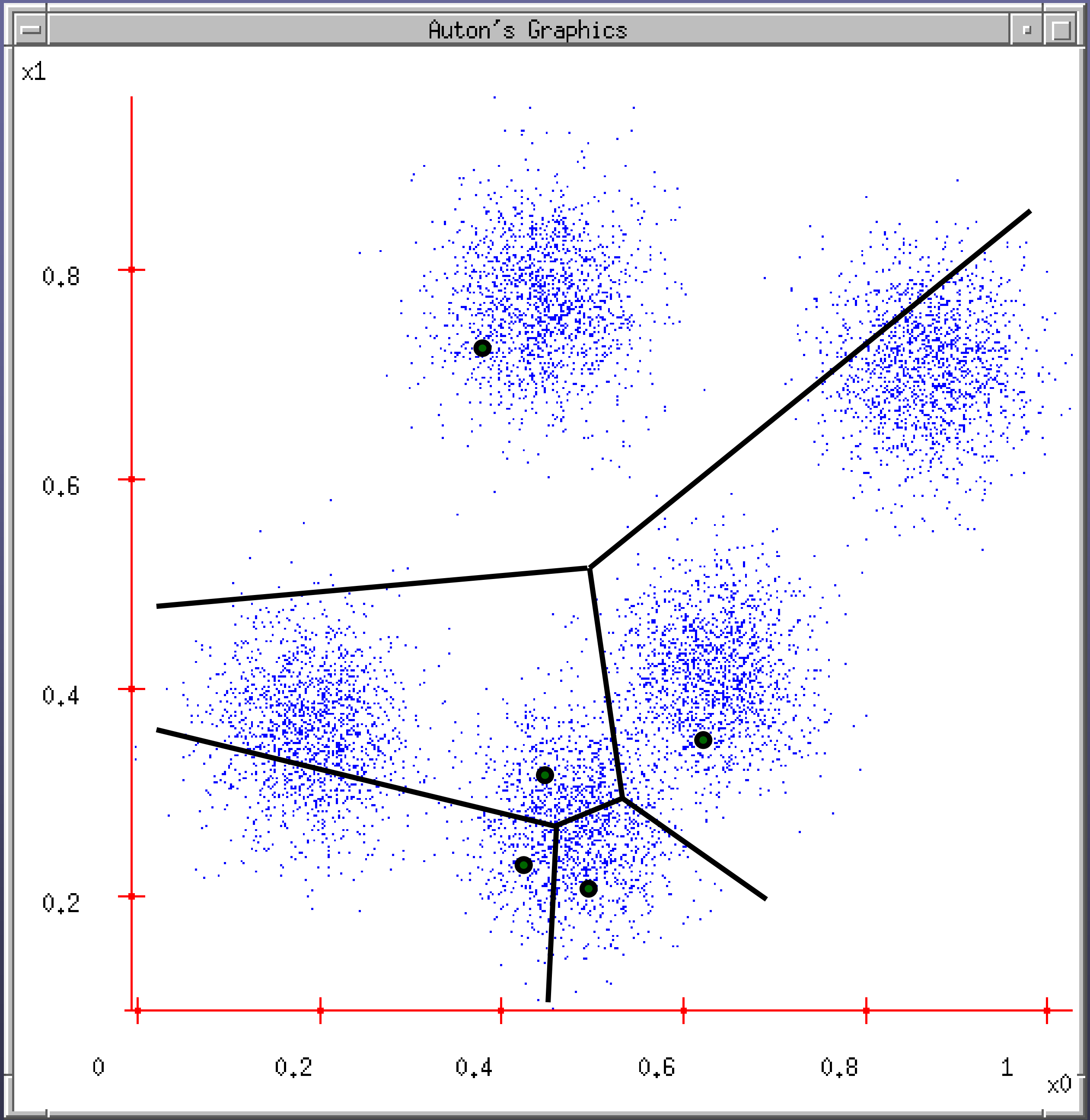
K-means

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



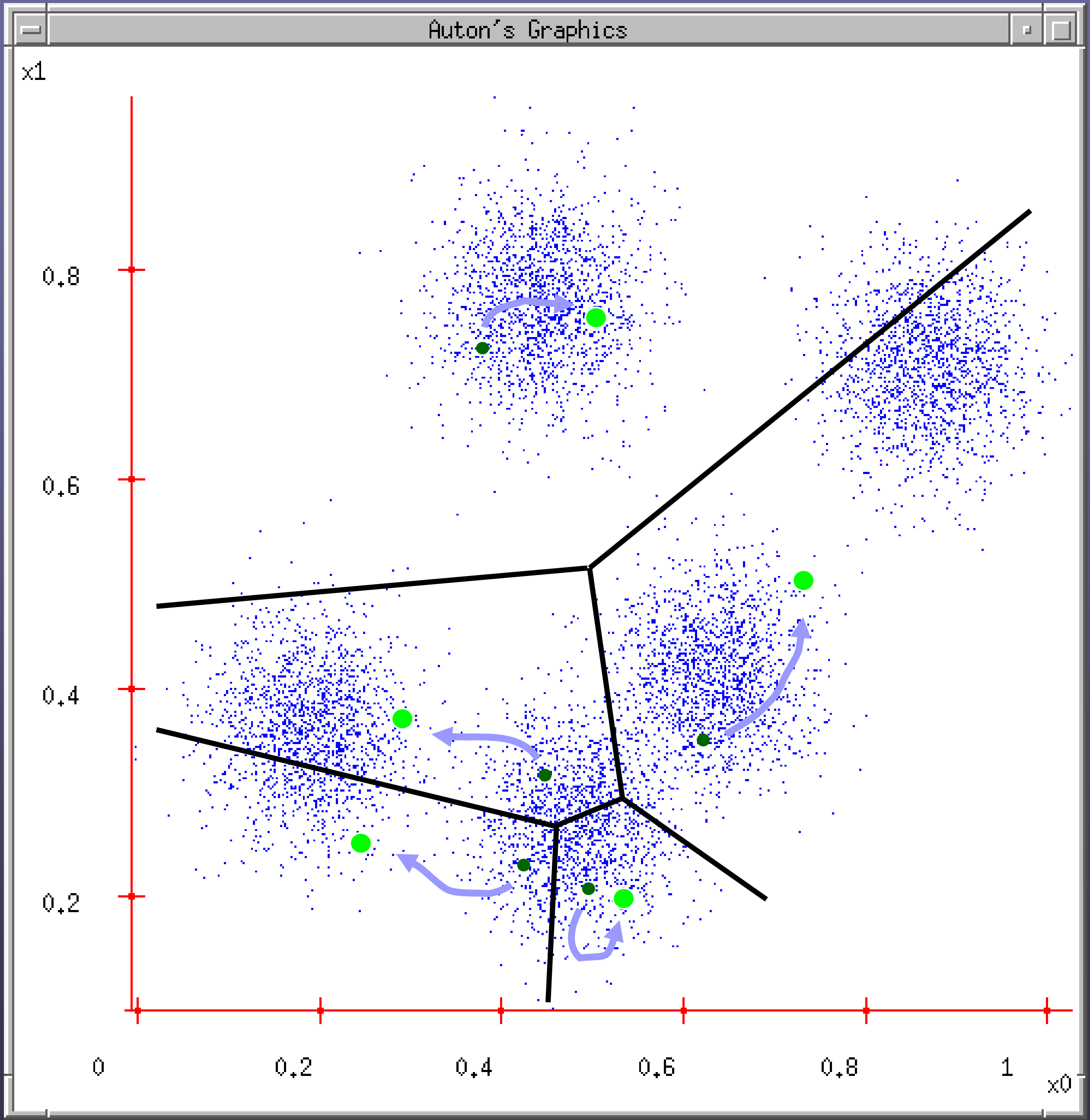
K-means

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. (Thus each Center "owns" a set of datapoints)



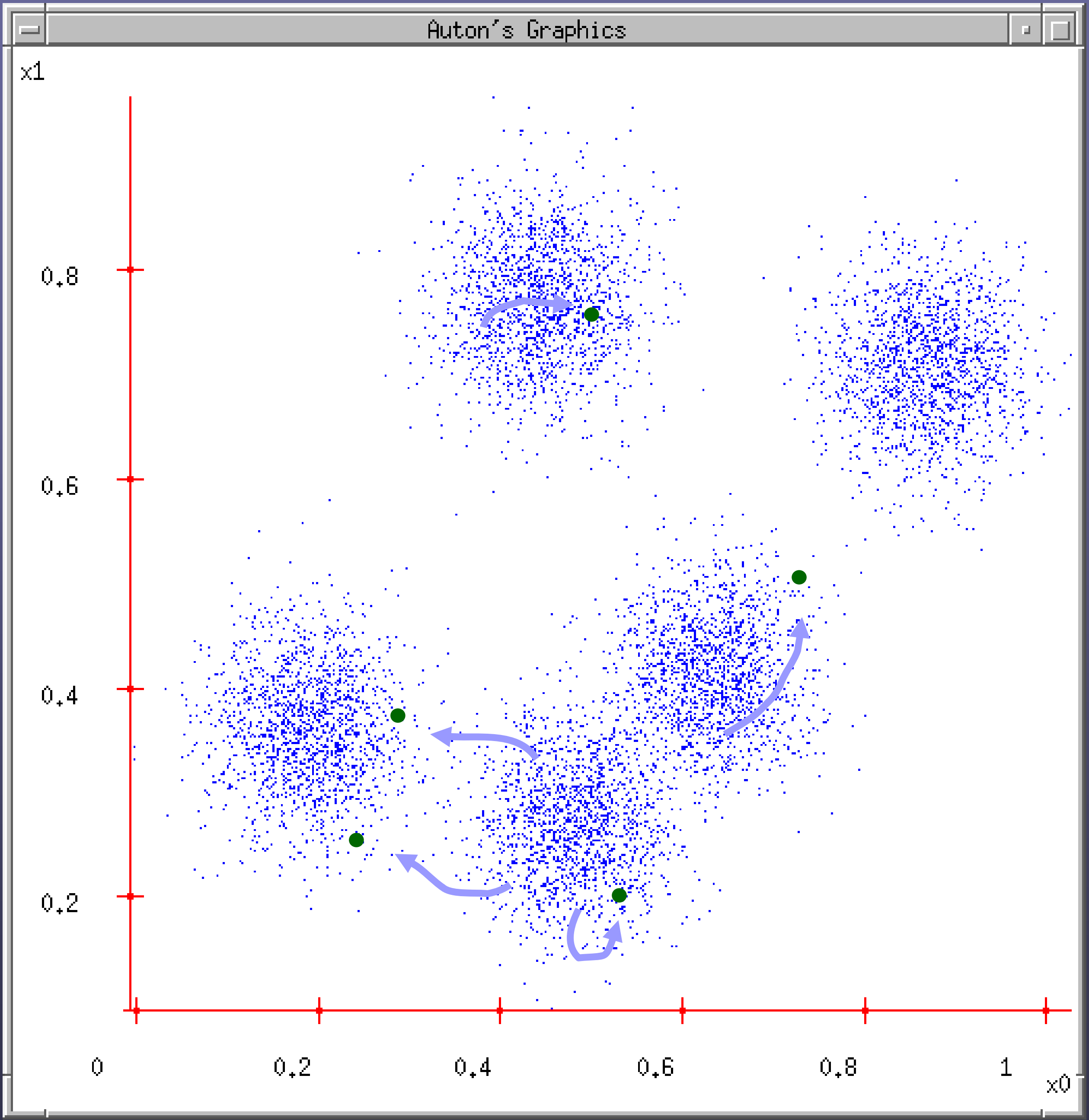
K-means

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



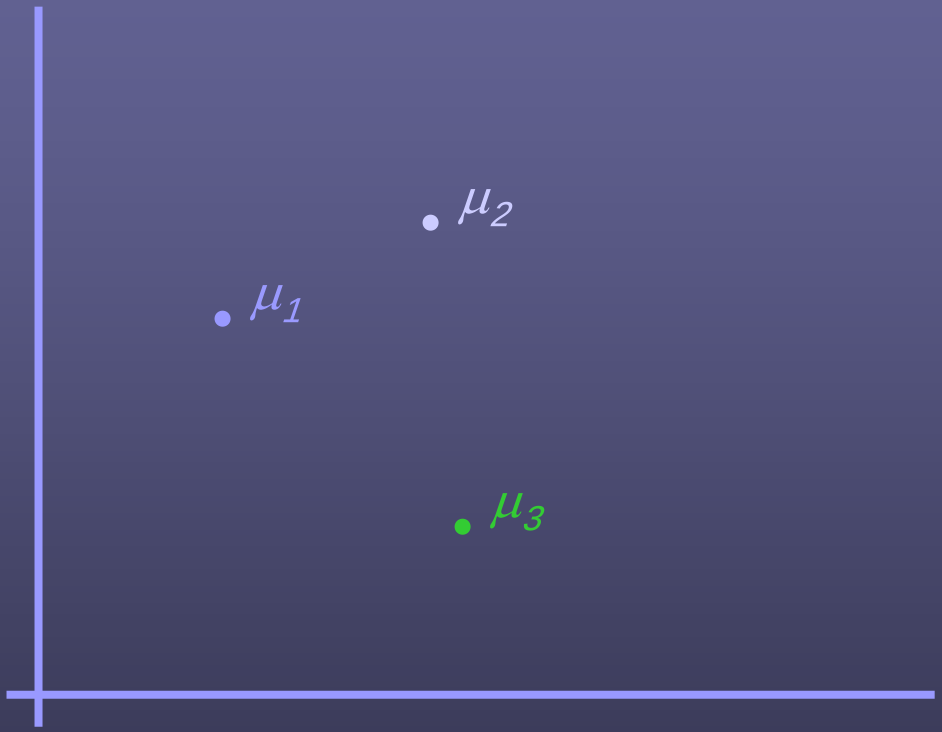
K-means

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!



The GMM assumption

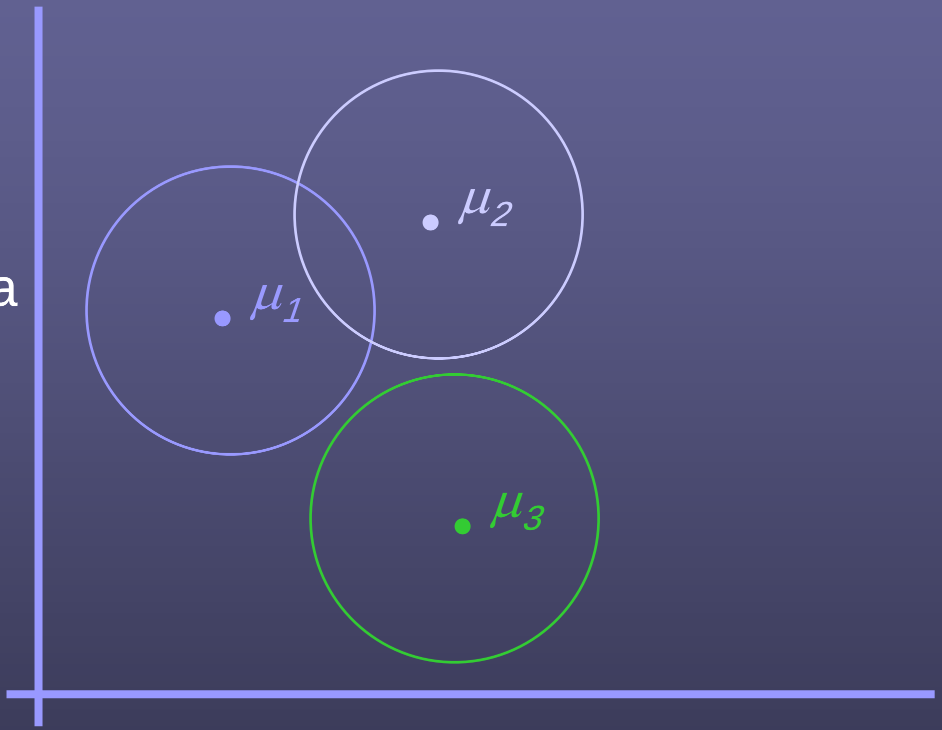
- There are k components. The i'th component is called ω_i
- Component ω_i has an associated mean vector μ_i



The GMM assumption

- There are k components. The i'th component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix $\sigma^2 \boldsymbol{I}$

Assume that each datapoint is generated according to the following recipe:

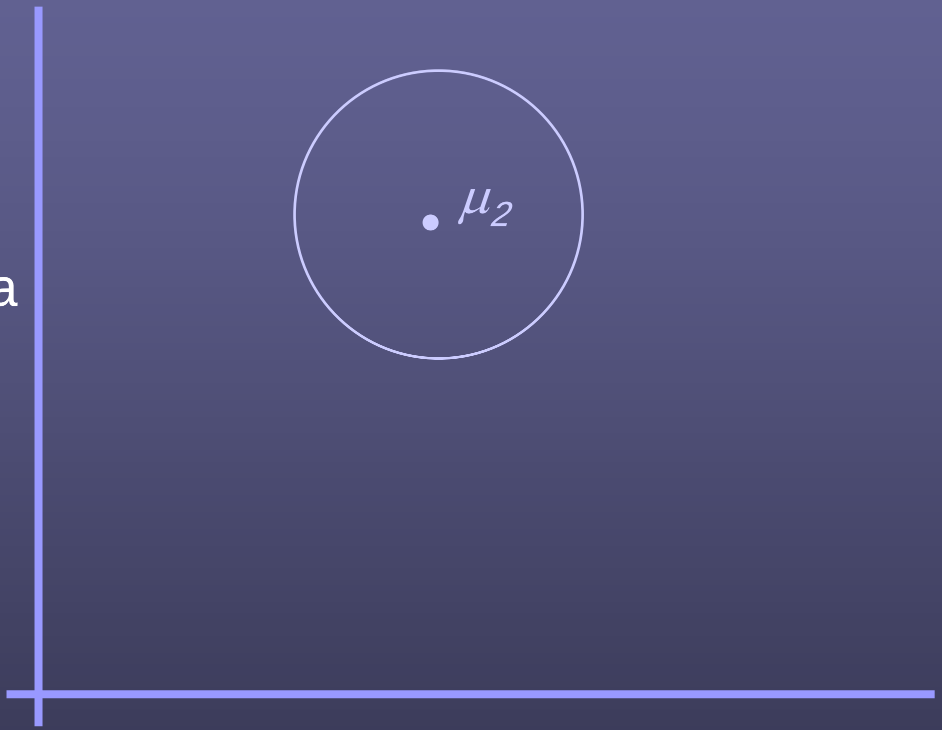


The GMM assumption

- There are k components. The i'th component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix $\sigma^2 \boldsymbol{I}$

Assume that each datapoint is generated according to the following recipe:

1. Pick a component at random. Choose component i with probability $P(\omega_i)$.

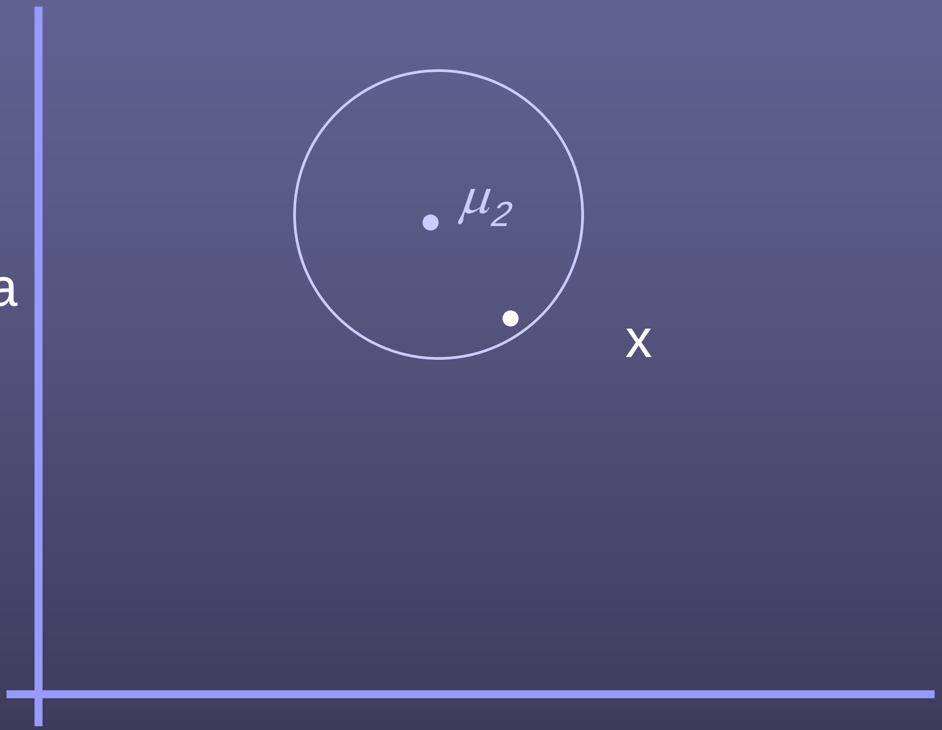


The GMM assumption

- There are k components. The i'th component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix $\sigma^2 \mathbf{I}$

Assume that each datapoint is generated according to the following recipe:

1. Pick a component at random. Choose component i with probability $P(\omega_i)$.
2. Datapoint ~ N(μ_i, $\sigma^2 \mathbf{I}$)

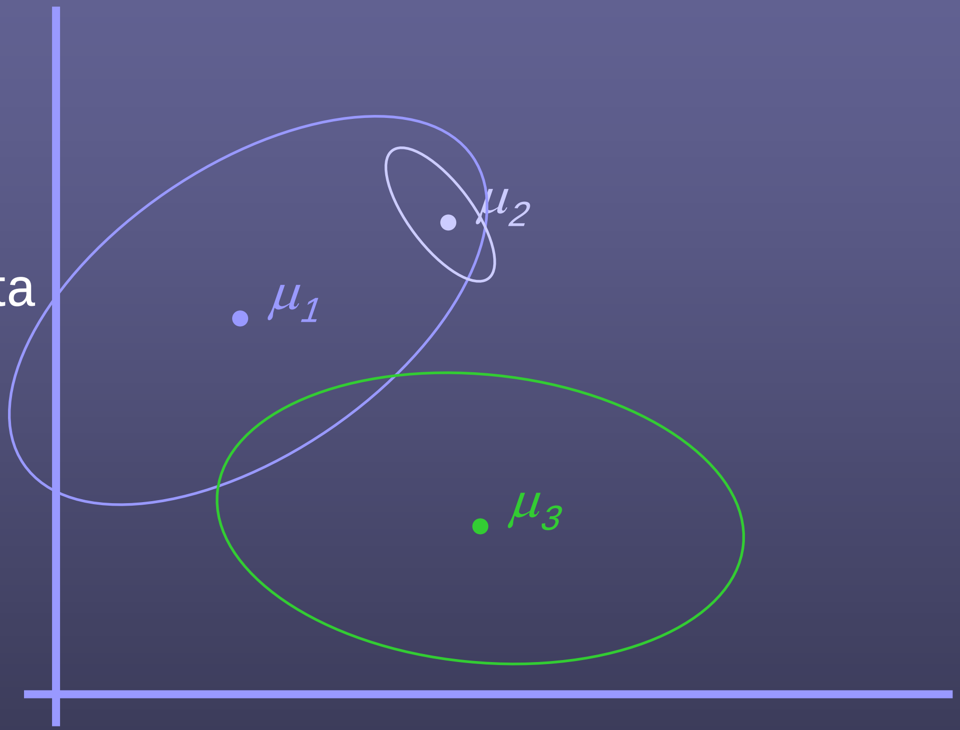


The General GMM assumption

- There are k components. The i'th component is called ω_i
- Component ω_i has an associated mean vector μ_i
- Each component generates data from a Gaussian with mean μ_i and covariance matrix Σ_i

Assume that each datapoint is generated according to the following recipe:

1. Pick a component at random. Choose component i with probability $P(\omega_i)$.
2. Datapoint ~ $N(\mu_i, \Sigma_i)$



Expectation-Maximization (EM)

- Solves estimation with incomplete data.
- Obtain initial estimates for parameters.
- Iteratively use estimates for missing data and continue until convergence.

EM - algorithm

- Iterative - algorithm
- Maximizing log-likelihood function

$$\mathcal{L} = \sum_{n=1}^{N} \log p(x^n)$$

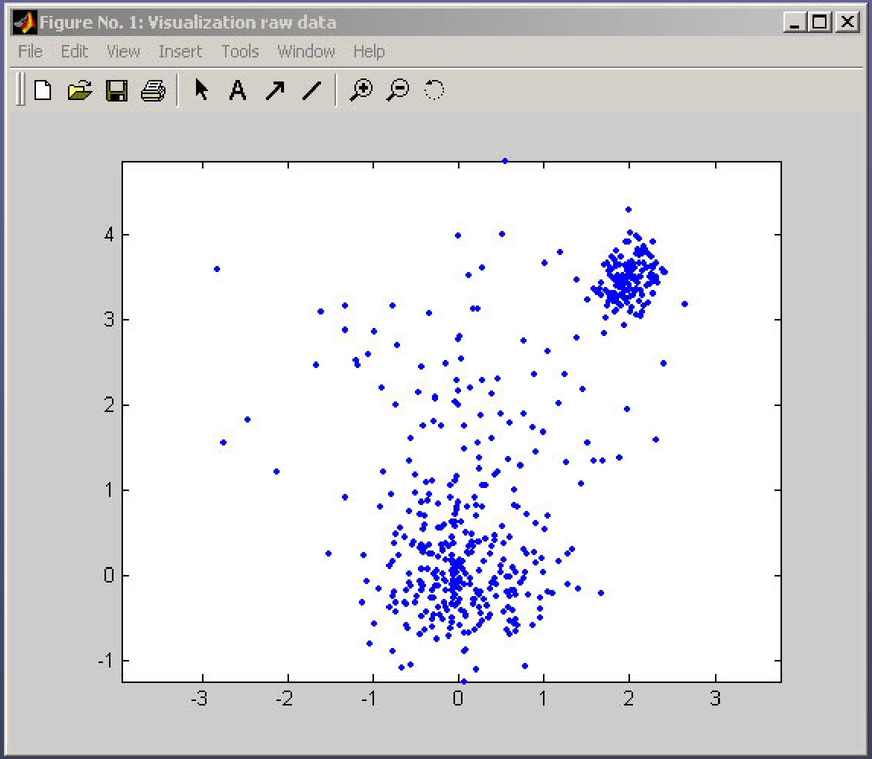
- E – step
- M – step

Sample 1

- Clustering data generated by a mixture of three Gaussians in 2 dimensions
  - number of points: 500
  - priors are: 0.3, 0.5 and 0.2
  - centers are: (2, 3.5), (0, 0), (0,2)
  - variances: 0.2, 0.5 and 1.0

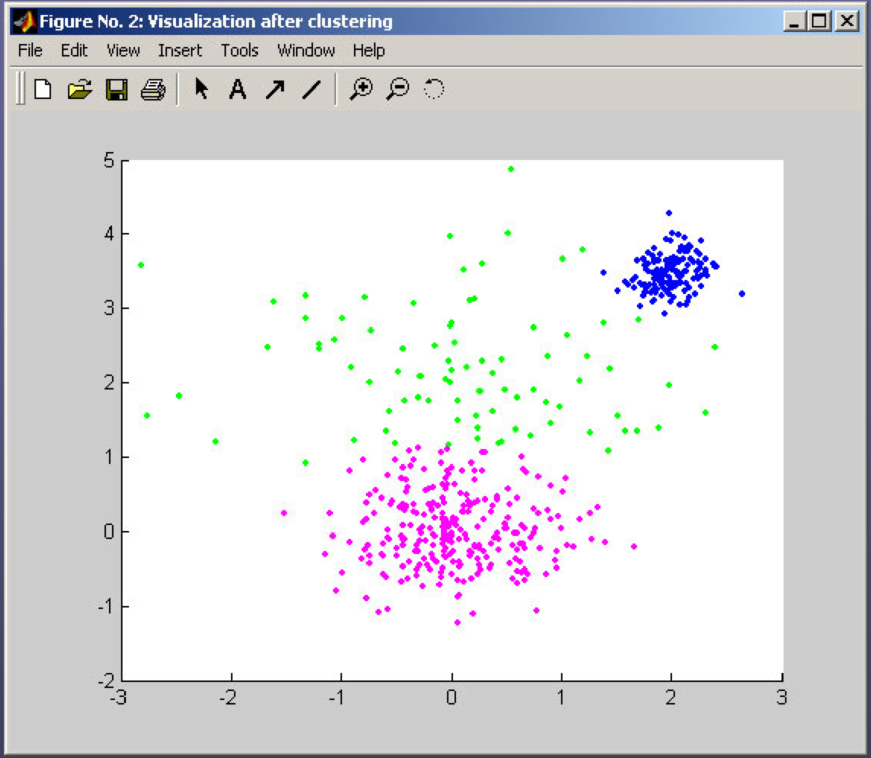
Sample 1

Raw data



- 150 (2, 3.5)
- 250 (0, 0)
- 100 (0,2)

After Clustering



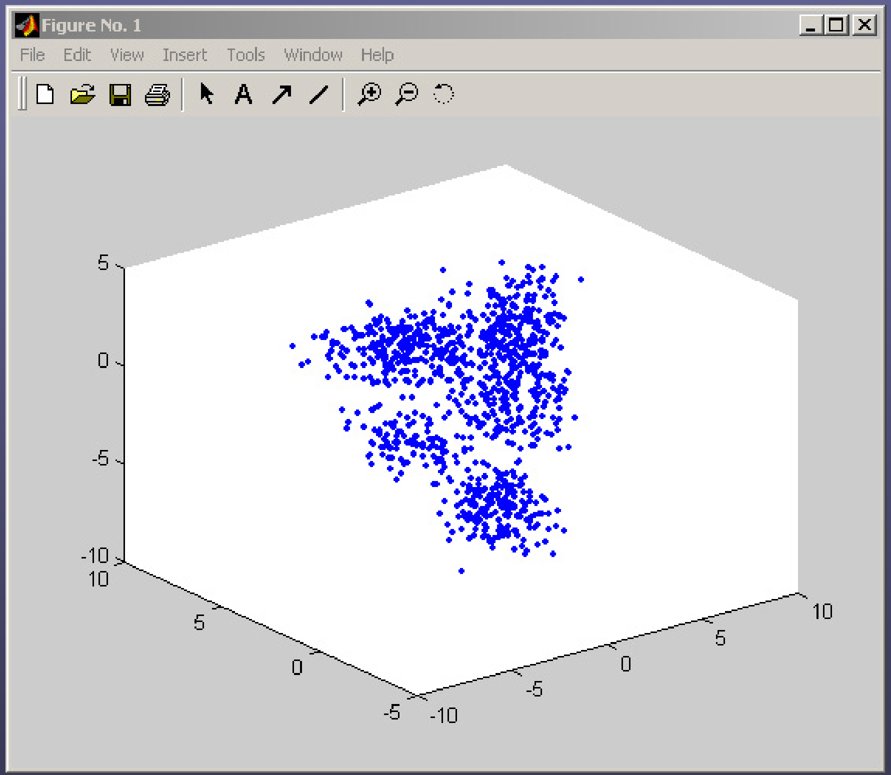
- 149 (1.9941, 3.4742)
- 265 (0.0306, 0.0026)
- 86 (0.1395, 1.9759)

Sample 2

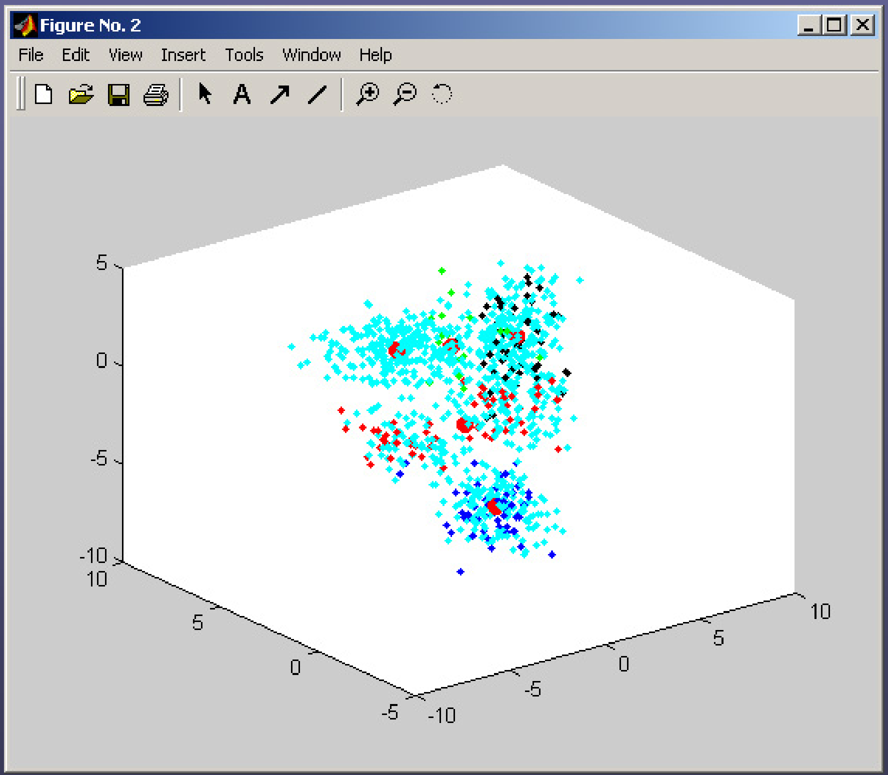
- Clustering three dimensional data
- Number of points:1000
- Unknown source
- Optimal number of components = ?
- Estimated parameters = ?

Sample 2

Raw data



After Clustering



Assumed number of clusters: 5

Sample 2 – table of estimated parameters

Number of components	4											
Components	1			2			3			4		
Priors	0.2495			0.1993			0.2507			0.3006		
Number of points	250			199			251			300		
Mean (x,y,z)	-0.8581	1.2983	-1.2396	2.4226	2.9885	-7.2448	4.9579	4.4317	0.1932	-3.1378	2.0412	2.8225
Covariance matrix	2.1520	-1.4307	1.2514	0.8694	0.0131	0.0529	0.2446	0.0961	-0.1539	3.5443	-0.0237	0.0787
	-1.4307	4.0383	-2.2968	0.0131	0.9721	0.0749	0.0961	1.3182	-0.9198	-0.0237	0.8773	0.2452
	1.2514	-2.2968	2.6001	0.0529	0.0749	0.9044	-0.1539	-0.9198	3.1431	0.0787	0.2452	0.3714

Number of components	5														
Components	1			2			3			4			5		
Priors	0.235			0.1993			0.2507			0.0925			0.2225		
Number of points	238			199			251			66			246		
Mean (x,y,z)	-0.8995	1.3222	-1.3665	2.4222	2.9883	-7.2442	4.9577	4.4311	0.1930	-1.1833	1.7887	2.4696	-3.7595	2.0728	2.8408
Covariance matrix	2.0741	-1.5238	1.2192	0.8697	0.0133	0.0517	0.2448	0.0967	-0.1538	2.6978	0.1347	-0.0558	2.4420	-0.0547	0.0724
	-1.5238	4.2281	-2.3959	0.0133	0.9720	0.0745	0.0967	1.3204	-0.9195	0.1347	0.9368	0.5855	-0.0547	0.9033	0.2076
	1.2192	-2.3959	2.4622	0.0517	0.0745	0.9062	-0.1538	-0.9195	3.1426	-0.0558	0.5855	0.9802	0.0724	0.2076	0.3132

Number of components	6																	
Components	1			2			3			4			5			6		
Priors	0.1482			0.1992			0.2502			0.086			0.2157			0.1008		
Number of points	148			199			250			69			233			101		
Mean (x,y,z)	-0.2010	-0.1246	-0.0452	2.4233	2.9890	-7.2462	4.9600	4.4393	0.1933	-1.3759	1.9430	2.7992	-3.8268	2.0611	2.8311	-1.7986	3.4215	-3.0350
Covariance matrix	2.0268	0.0183	0.2065	0.8691	0.0124	0.0549	0.2431	0.0890	-0.1544	2.1632	0.1963	0.2091	2.3922	-0.0809	0.0489	0.9676	-0.1883	-0.0290
	0.0183	1.0671	0.3169	0.0124	0.9723	0.0765	0.0890	1.2947	-0.9220	0.1963	0.8568	0.3335	-0.0809	0.9284	0.2156	-0.1883	0.8868	0.2724
	0.2065	0.3169	0.4099	0.0549	0.0765	0.9006	-0.1544	-0.9220	3.1496	0.2091	0.3335	0.4665	0.0489	0.2156	0.3299	-0.0290	0.2724	0.3562

References

- [1] http://www.autonlab.org/tutorials/gmm14.pdf
- [2] http://www.autonlab.org/tutorials/kmeans11.pdf
- [3] http://info.ilab.sztaki.hu/~lukacs/AdatbanyaEA2005/klaszterezes.pdf
- [4] http://www.stat.auckland.ac.nz/~balemi/Data%20Mining%20in%20Market%20Research.ppt
- [5] http://www.ncrg.aston.ac.uk/netlab