CS 461: Machine Learning
Lecture 7

Dr. Kiri Wagstaff
wkiri@wkiri.com

Plan for Today

- Unsupervised Learning
- K-means Clustering
- EM Clustering

- Homework 4

Review from Lecture 6

- Parametric methods
  - Data comes from distribution
  - Bernoulli, Gaussian, and their parameters
  - How good is a parameter estimate? (bias, variance)
- Bayes estimation
  - ML: use the data (assume equal priors)
  - MAP: use the prior and the data
- Parametric classification
  - Maximize the posterior probability

Clustering

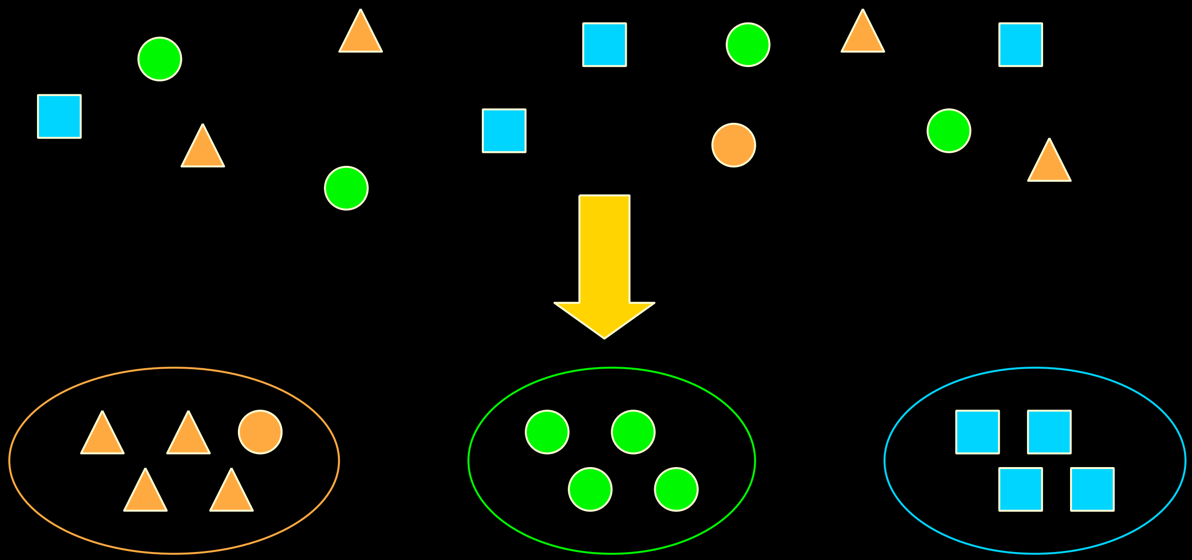
Chapter 7

Unsupervised Learning

- The data has no labels!
- What can we still learn?
  - Salient groups in the data
  - Density in feature space
- Key approach: clustering
- … but also:
  - Association rules
  - Density estimation
  - Principal components analysis (PCA)

Clustering

- Group items by similarity



- Density estimation, cluster models

Applications of Clustering

- Image Segmentation

[Ma and Manjunath, 2004]

[Selim Aksoy]

- Data Mining: Targeted marketing
- Remote Sensing: Land cover types
- Text Analysis

Applications of Clustering

- Text Analysis: Noun Phrase Coreference

Input text

John Simon, Chief Financial Officer of Prime Corp. since 1986, saw his pay jump 20%, to $1.3 million, as the 37-year-old also became the financial-services company's president.

→

Cluster JS

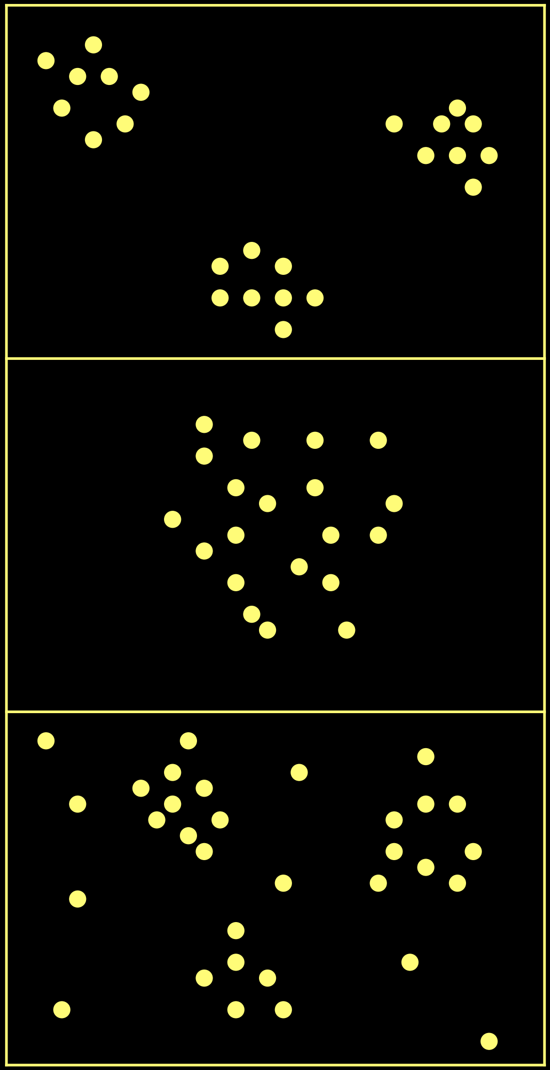
John Simon
Chief Financial Officer
his
the 37-year-old
president

Cluster PC

Prime Corp.
the financial-services company

Singletons

1986
pay
20%
$1.3 million



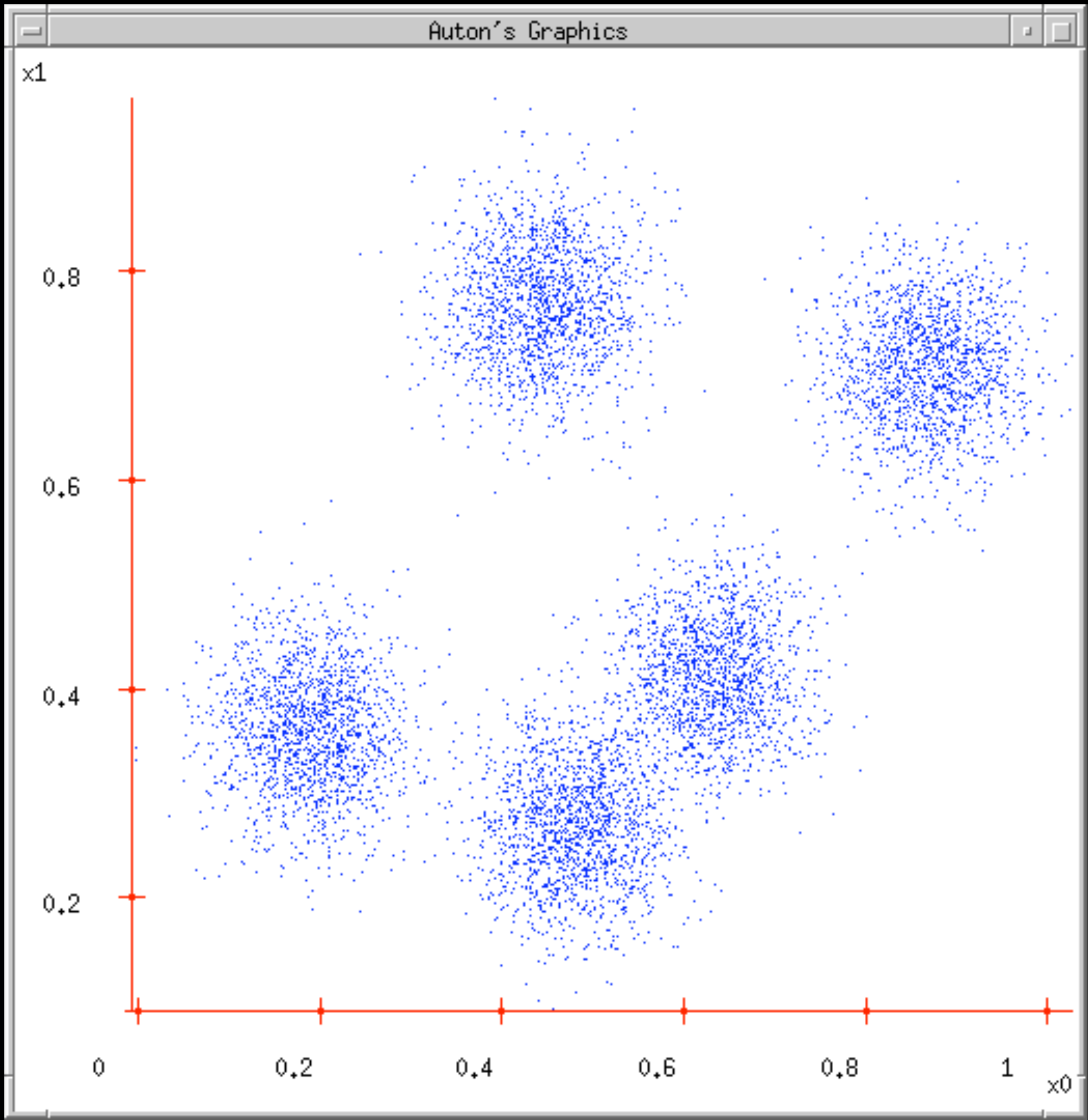
Sometimes easy

Sometimes impossible

and sometimes
in between

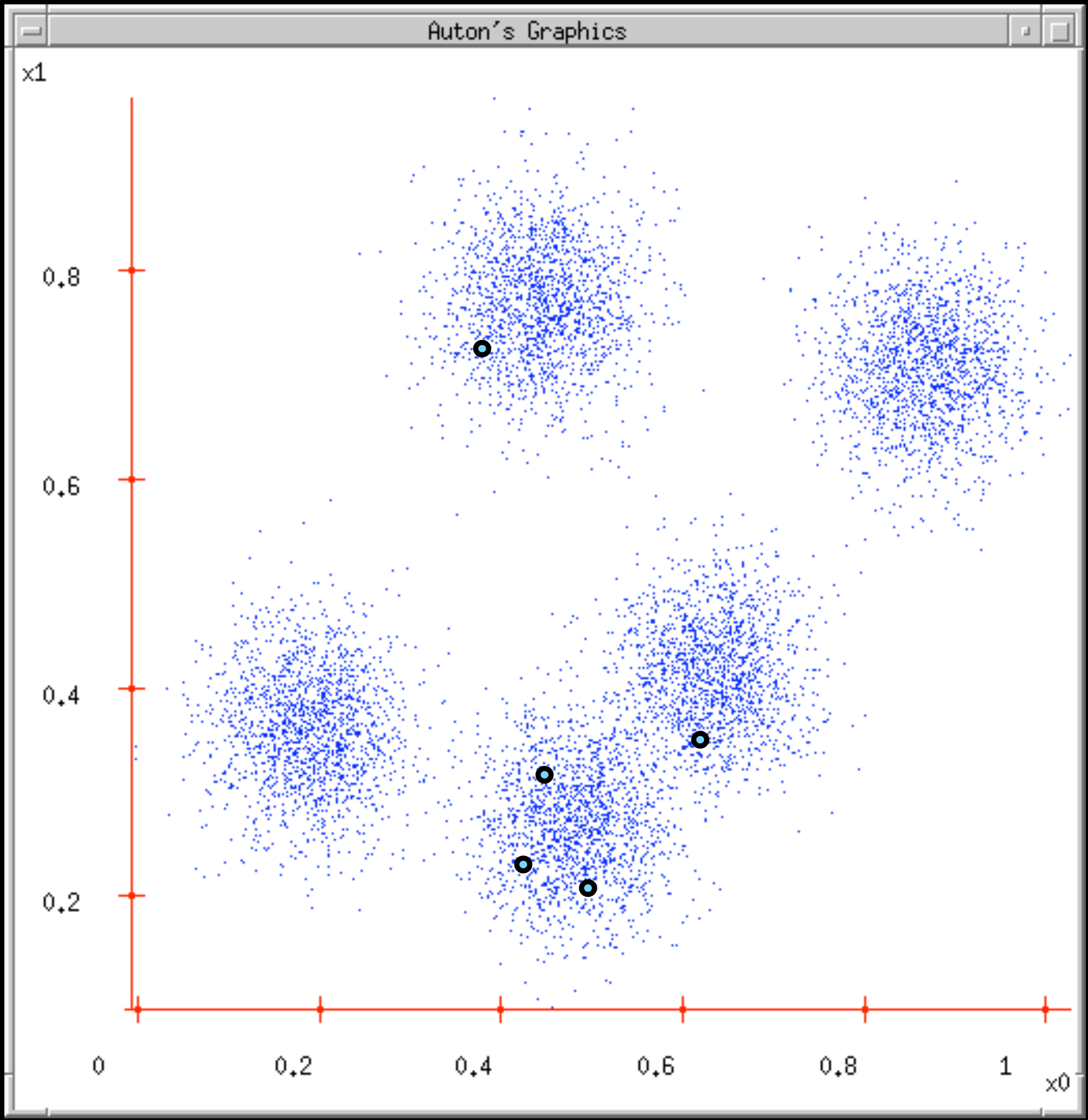
K-means

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



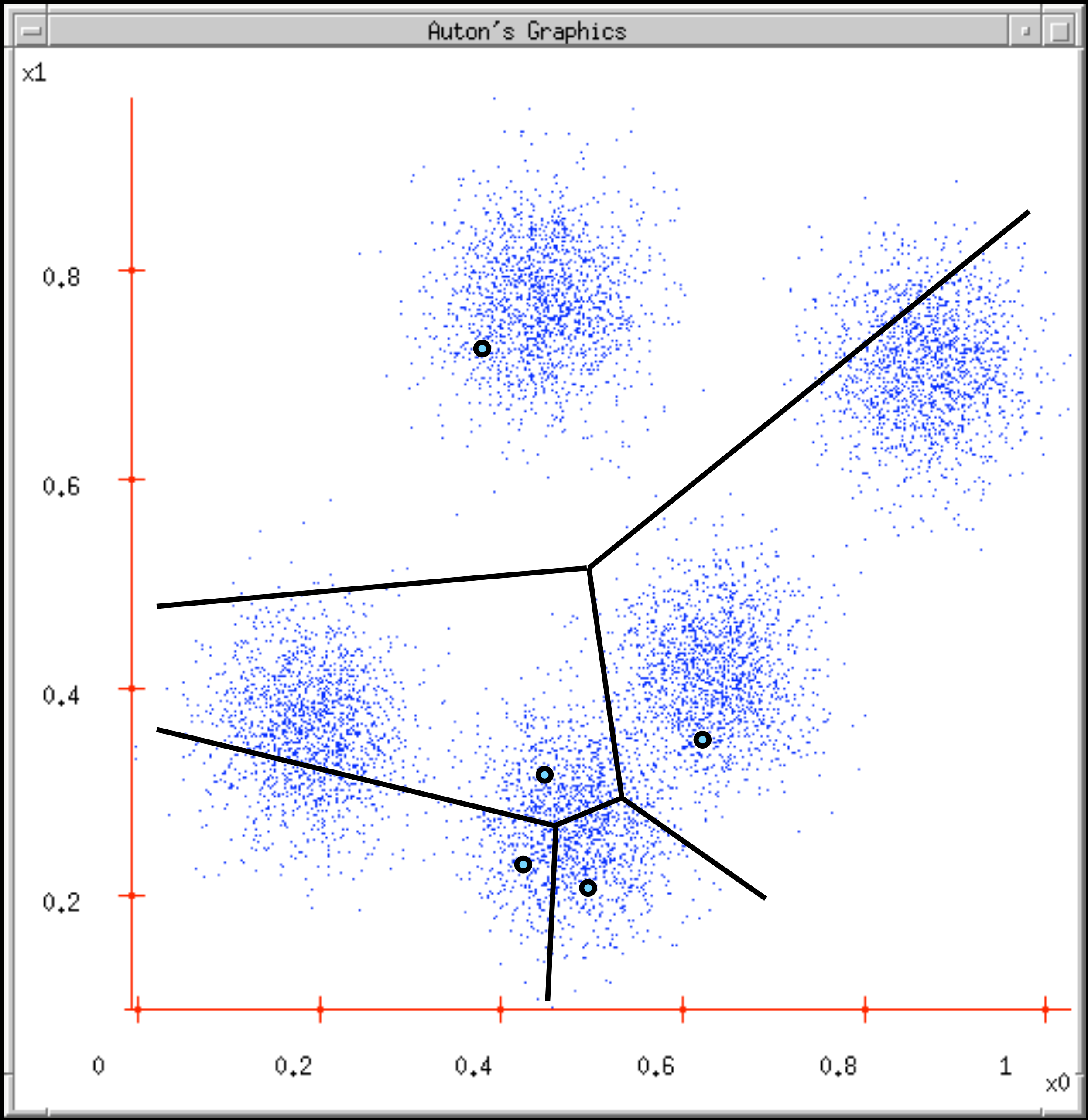
K-means

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



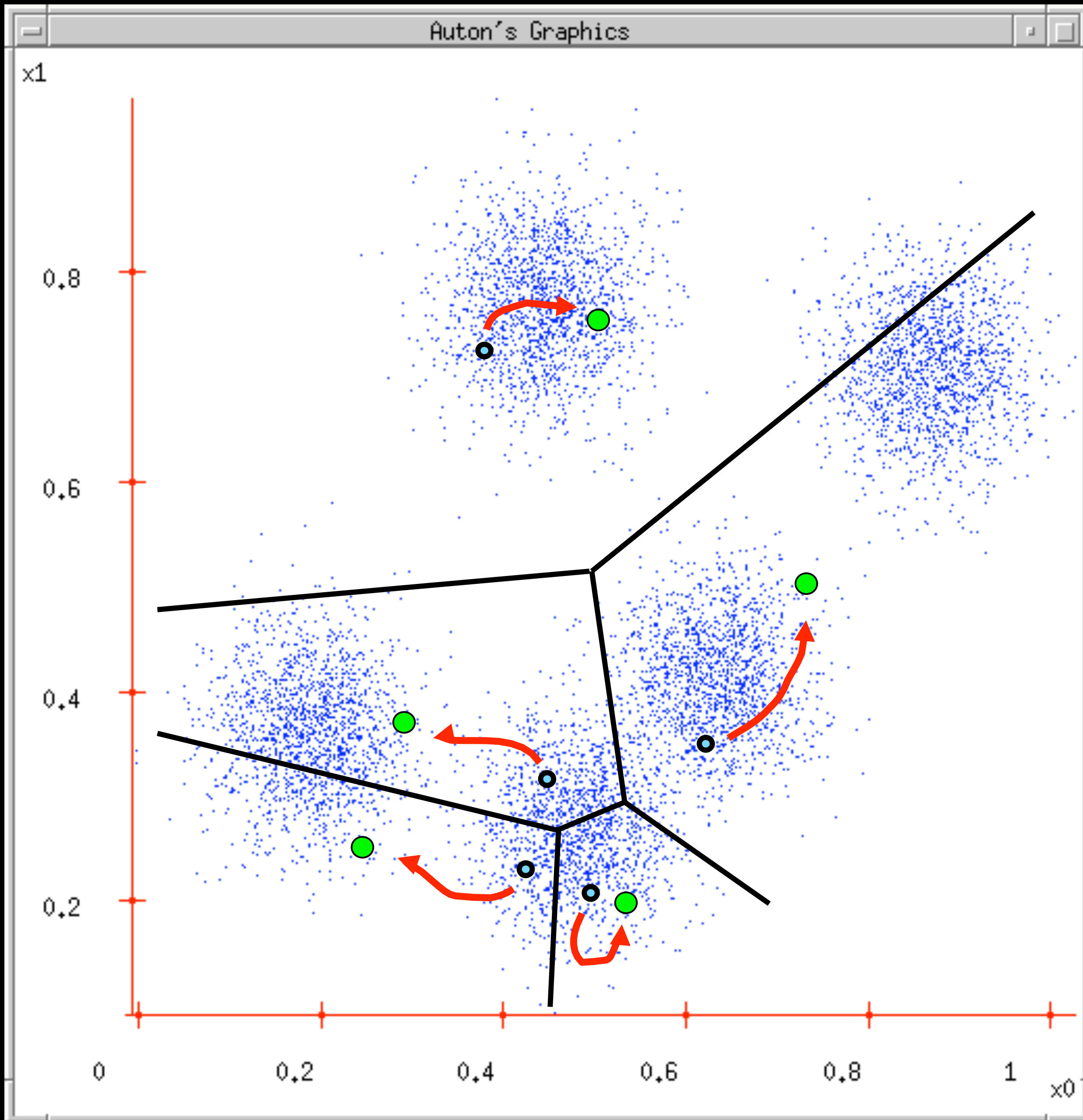
K-means

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)



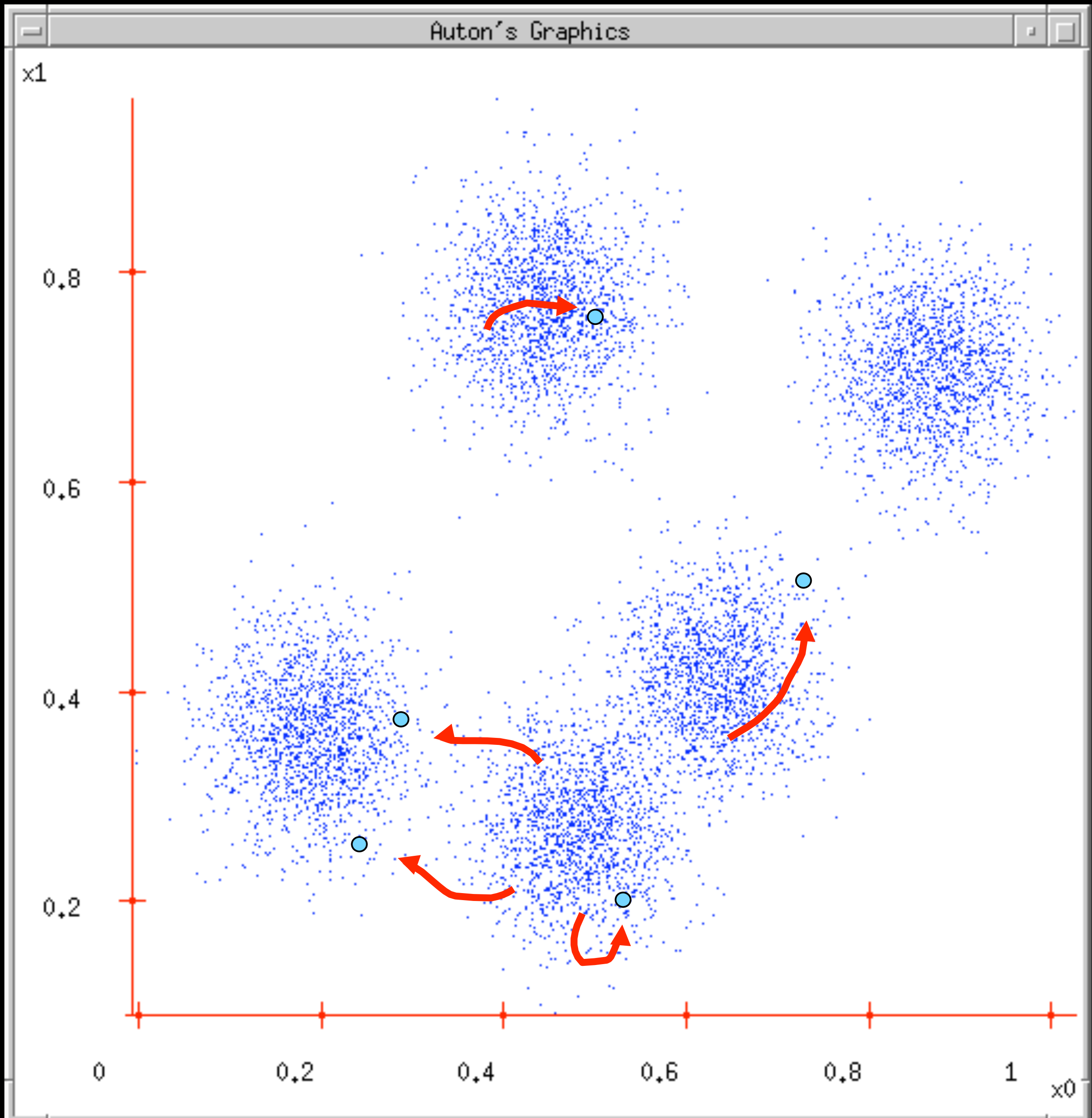
K-means

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

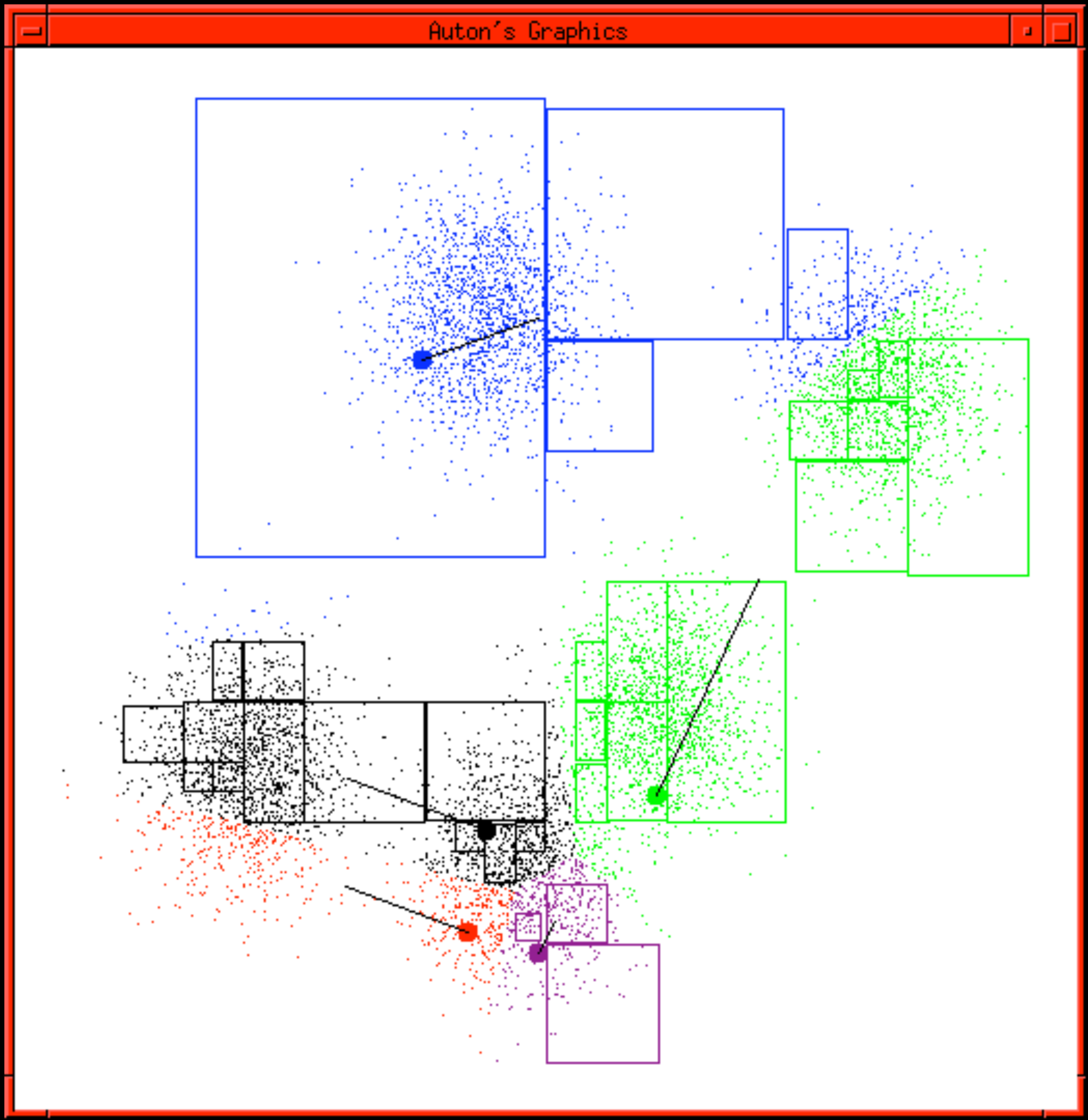
1. Ask user how many clusters they'd like. *(e.g. k=5)*
2. Randomly guess k cluster Center locations
3. Each datapoint finds out which Center it's closest to.
4. Each Center finds the centroid of the points it owns…
5. …and jumps there
6. …Repeat until terminated!



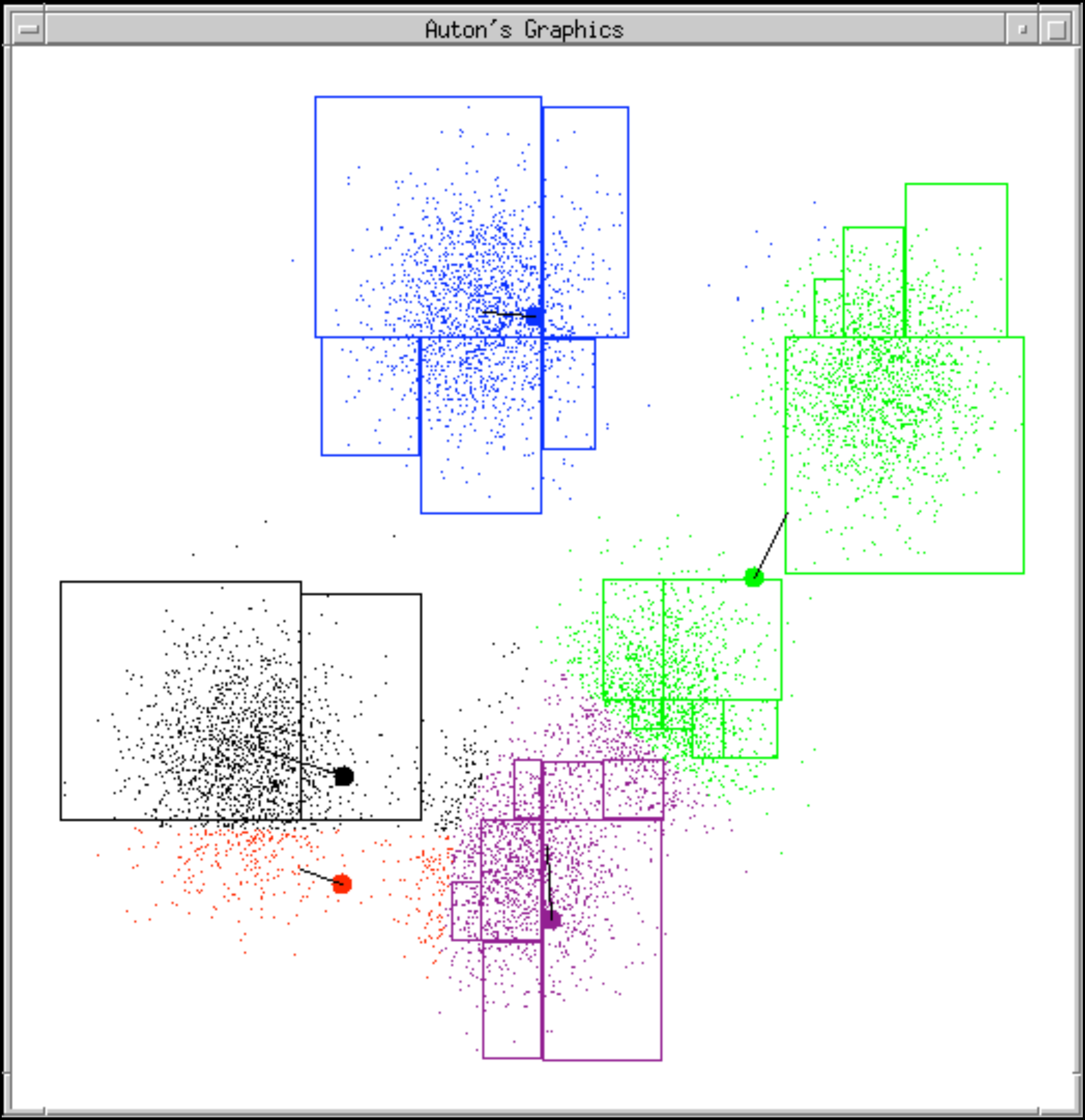
K-means
Start: k=5

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

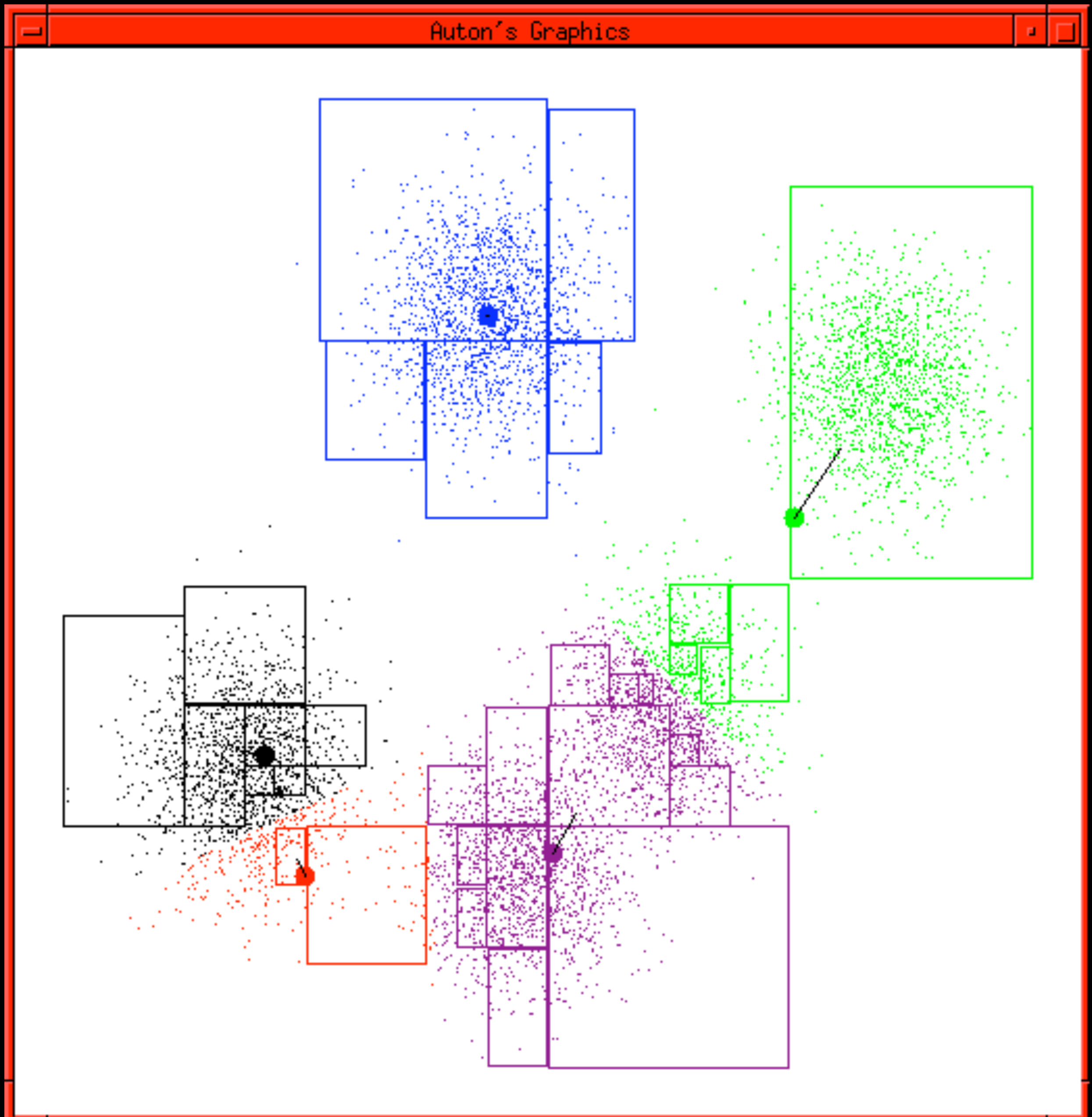
Dan Pelleg and Andrew Moore. Accelerating Exact k-means Algorithms with Geometric Reasoning. Proc. Conference on Knowledge Discovery in Databases 1999, (KDD99) (available on www.autonlab.org/pap.html*)*



K-means continues…

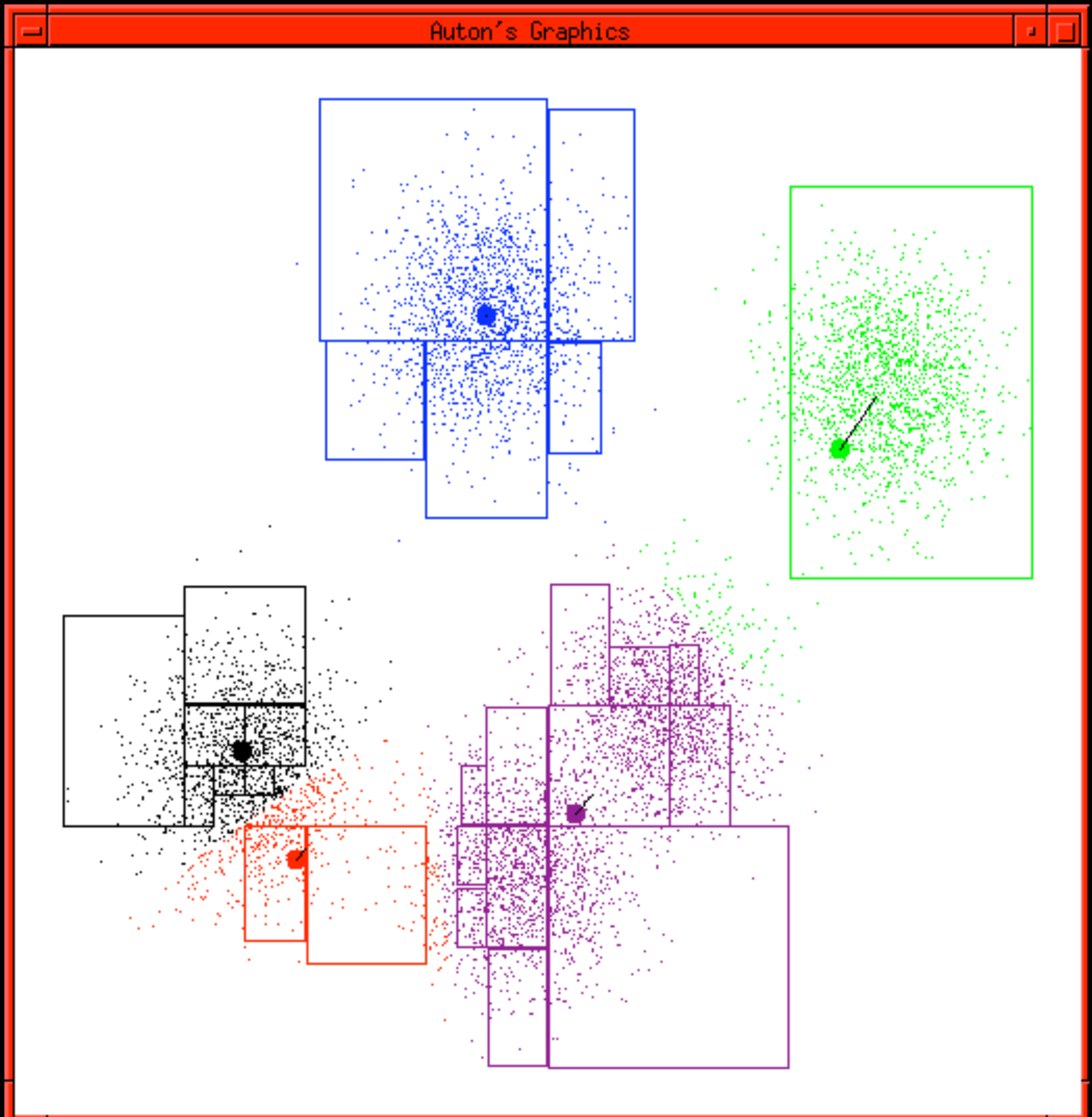


K-means continues…

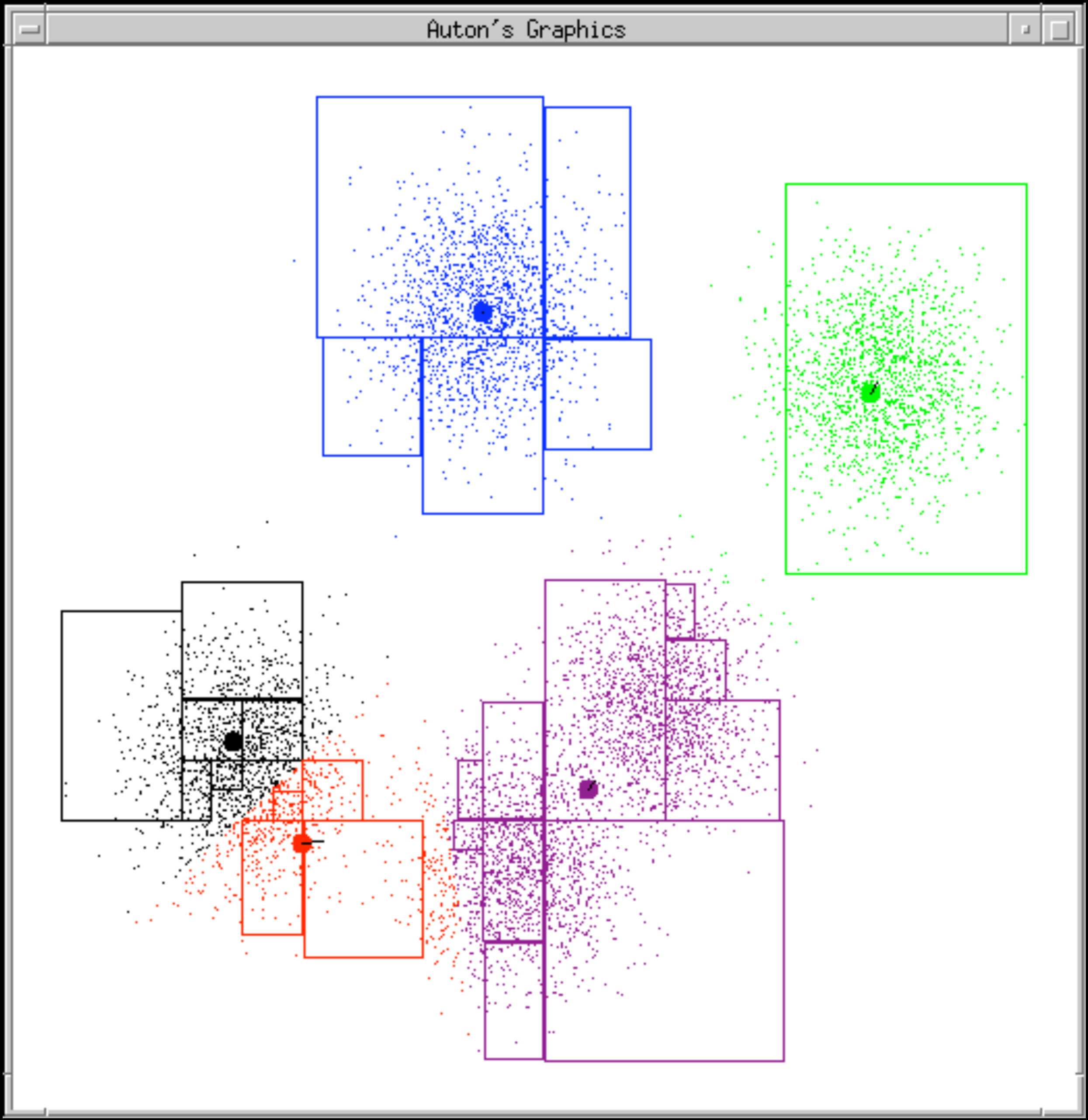


[© Andrew Moore]

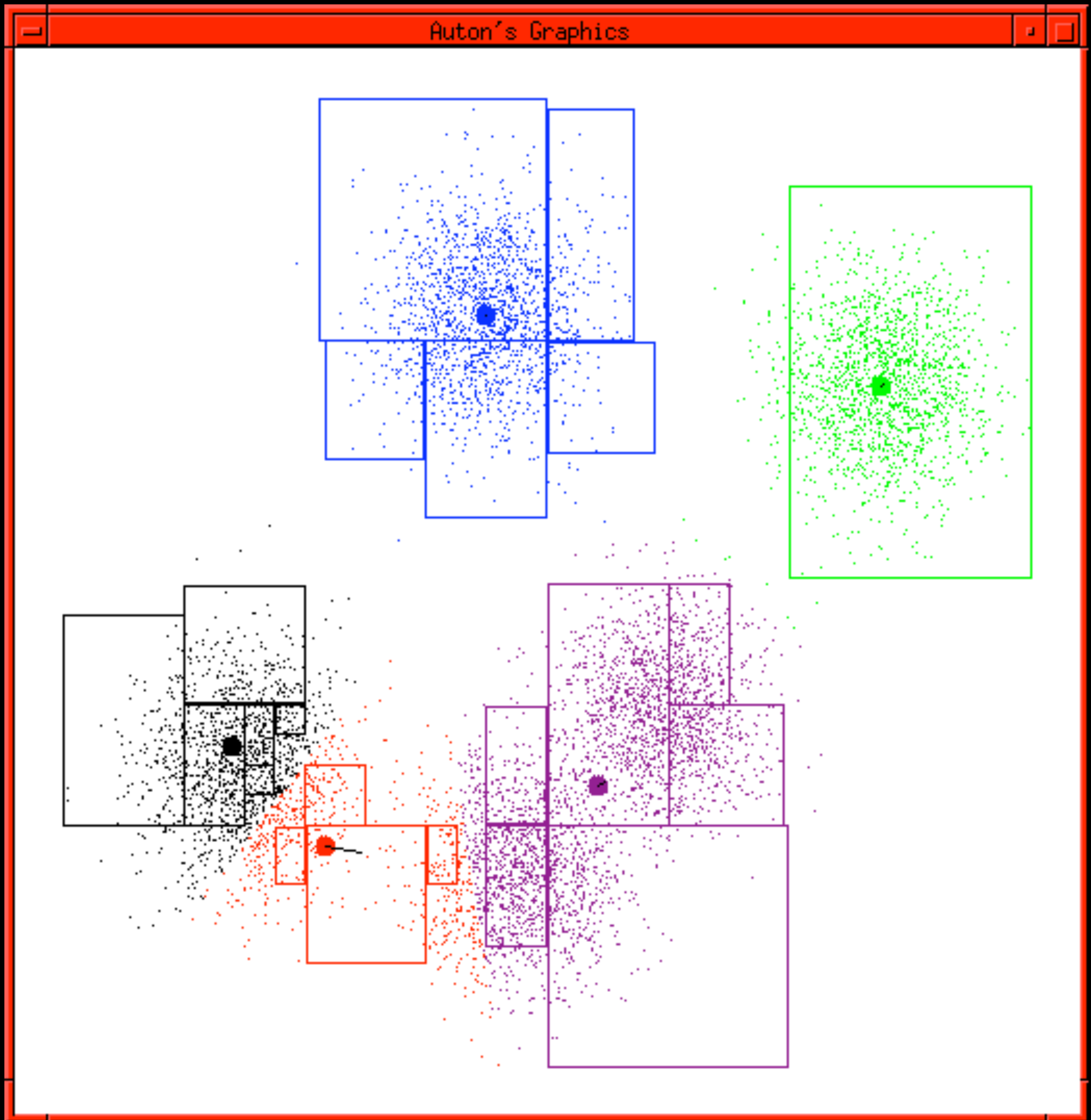
K-means continues…



K-means continues…

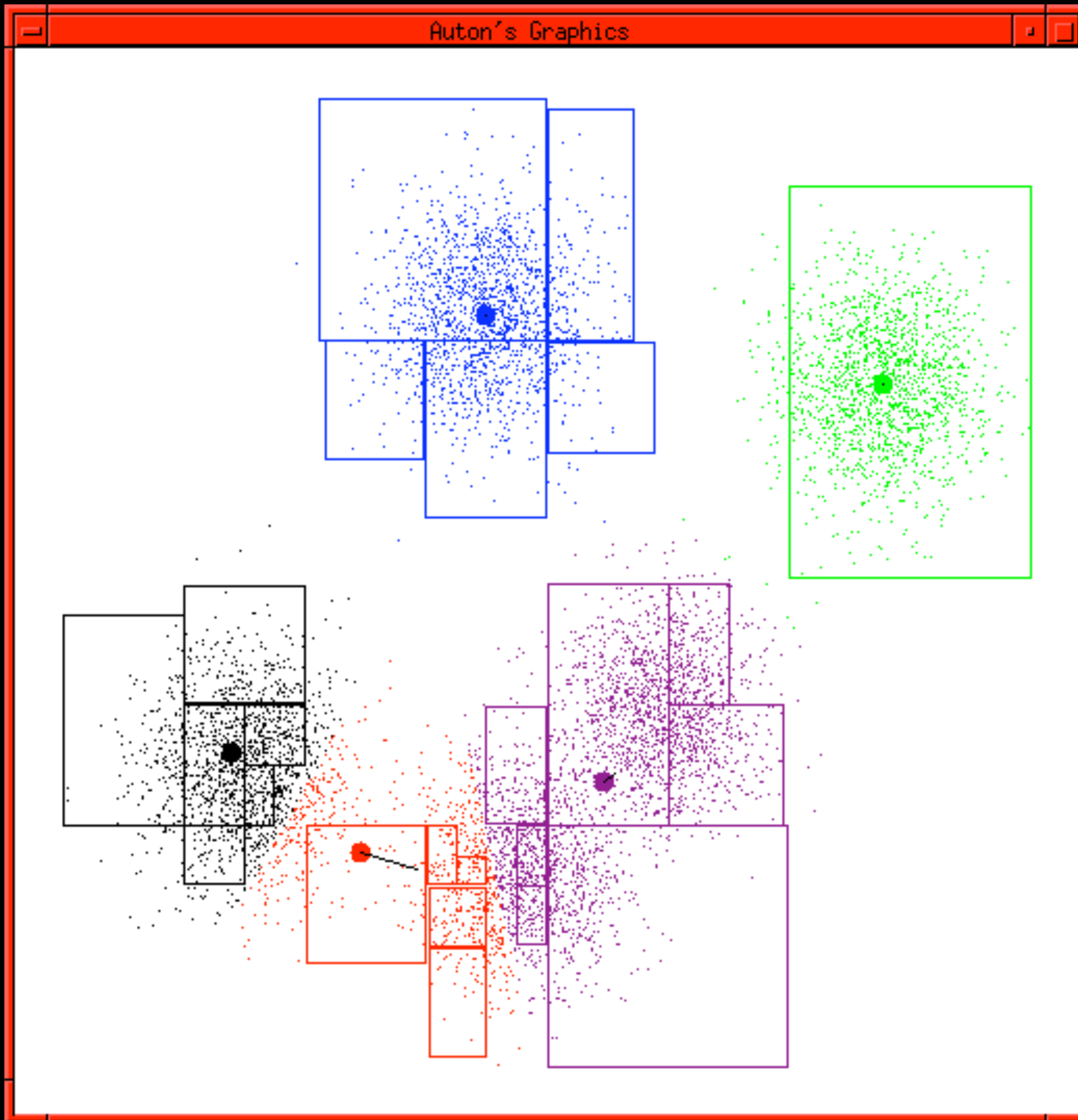


K-means continues…

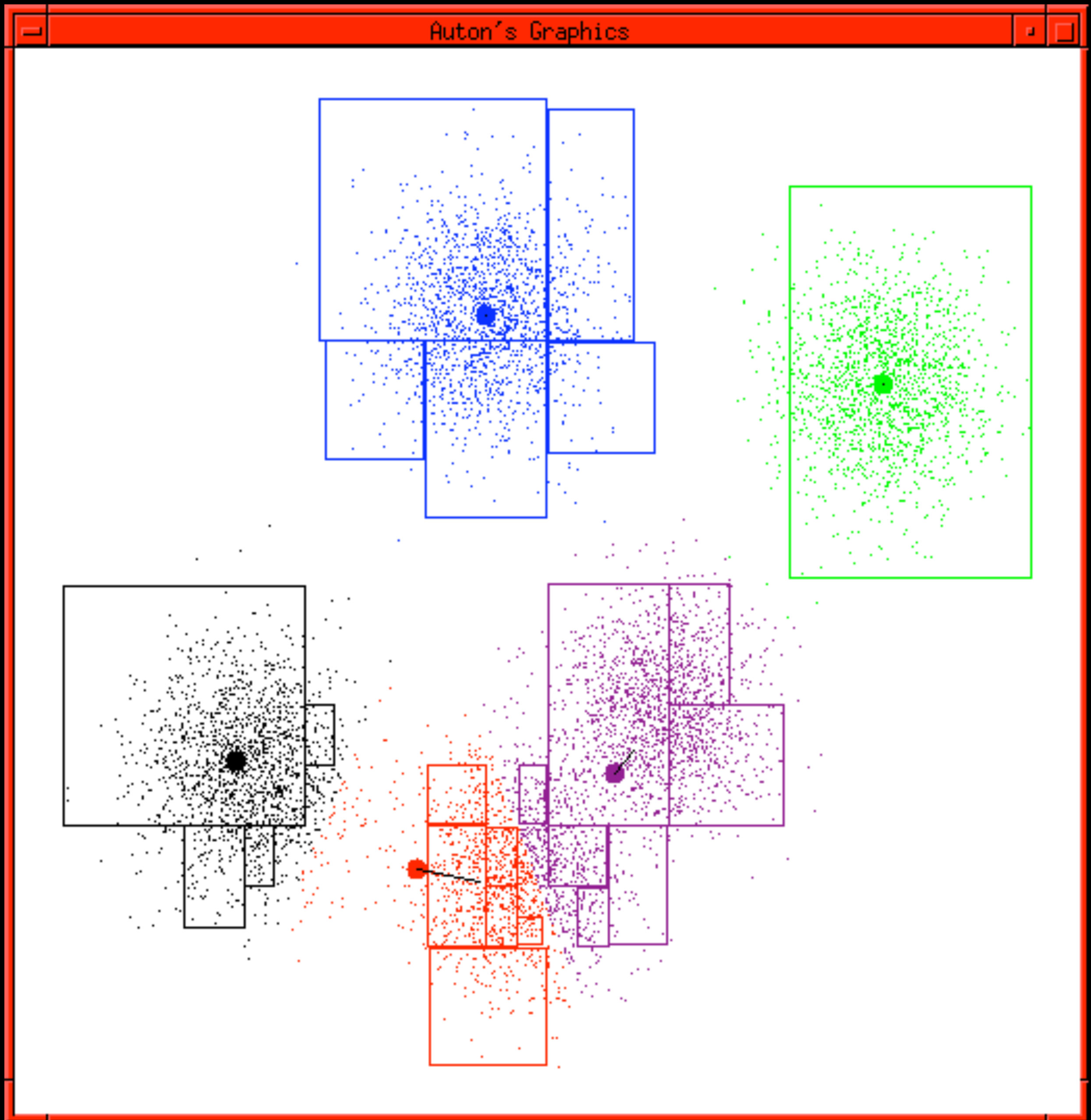


[© Andrew Moore]

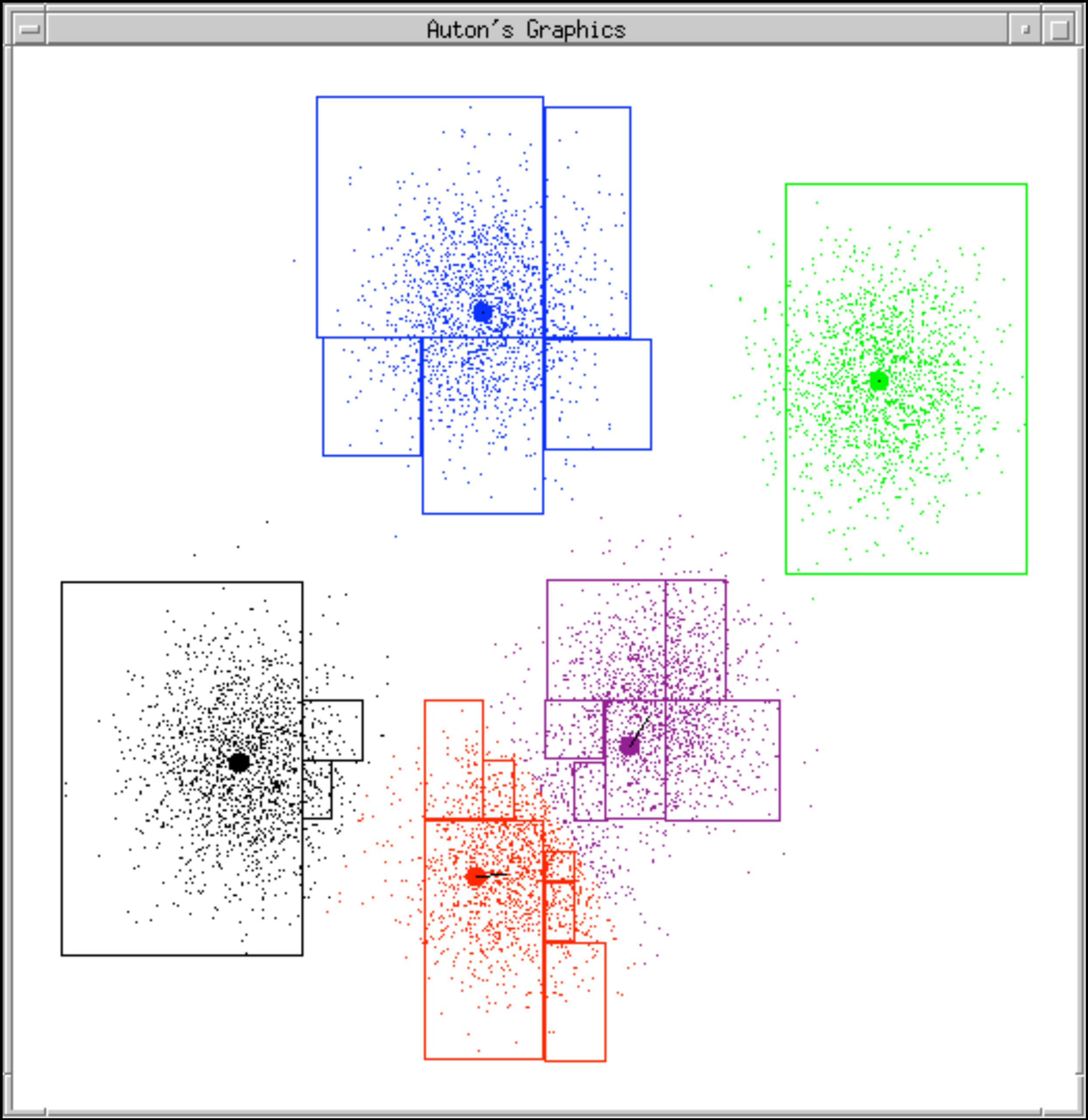
K-means continues…



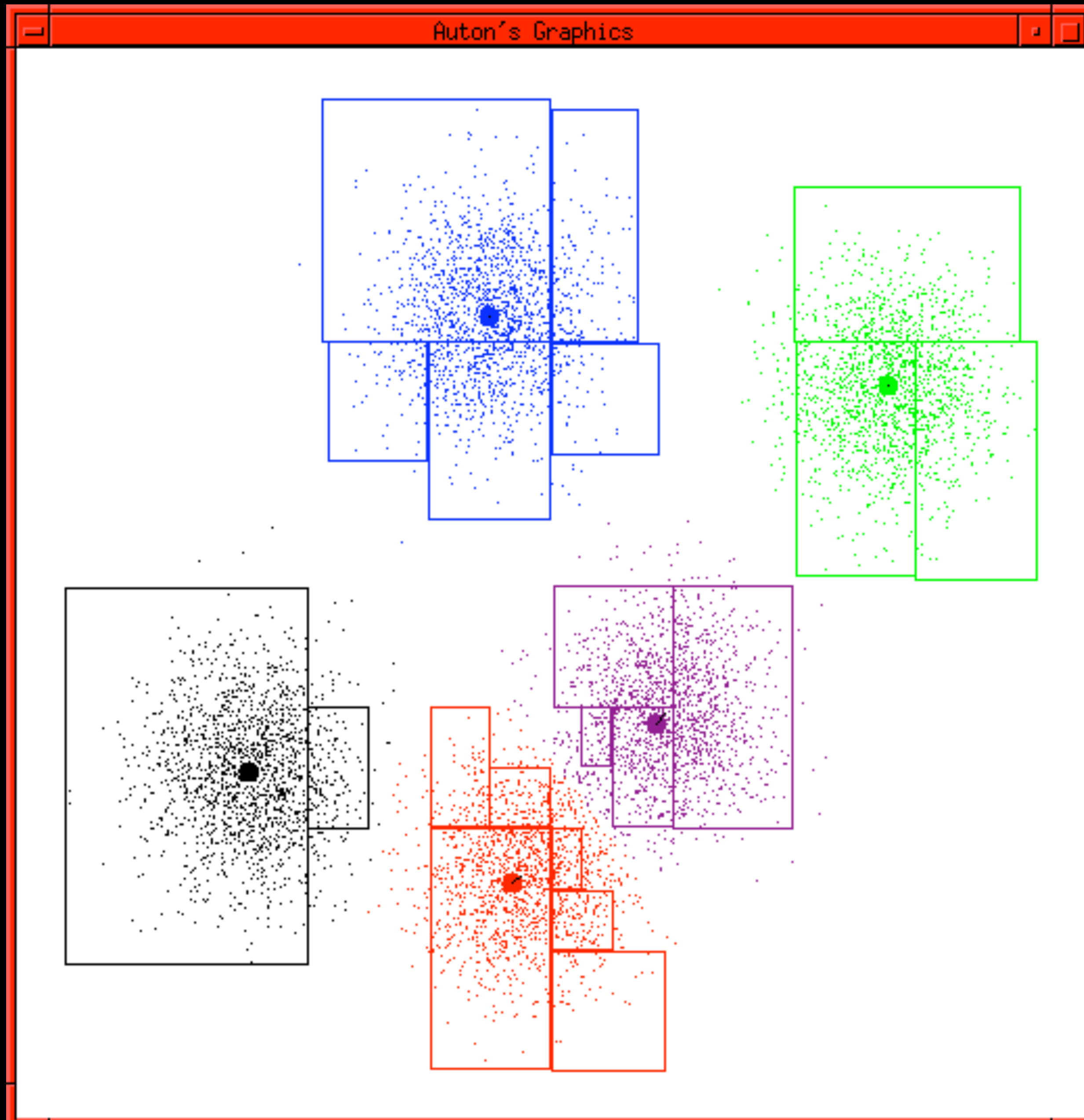
K-means continues…



K-means continues…



K-means terminates



K-means Algorithm

1. Randomly select *k* cluster centers
2. While (points change membership)
   1. Assign each point to its closest cluster
      - (Use your favorite distance metric)
   2. Update each center to be the mean of its items

- Objective function: Variance

$$V = \sum_{c=1}^{k} \sum_{x_j \in C_c} dist(x_j, \mu_c)^2$$

- K-means applet

K-means Algorithm: Example

1. Randomly select *k* cluster centers
2. While (points change membership)
   1. Assign each point to its closest cluster
      - (Use your favorite distance metric)
   2. Update each center to be the mean of its items

- Objective function: Variance

$$V = \sum_{c=1}^{k} \sum_{x_j \in C_c} dist(x_j, \mu_c)^2$$

- Data: [1, 15, 4, 2, 17, 10, 6, 18]

K-means for Compression

Original image

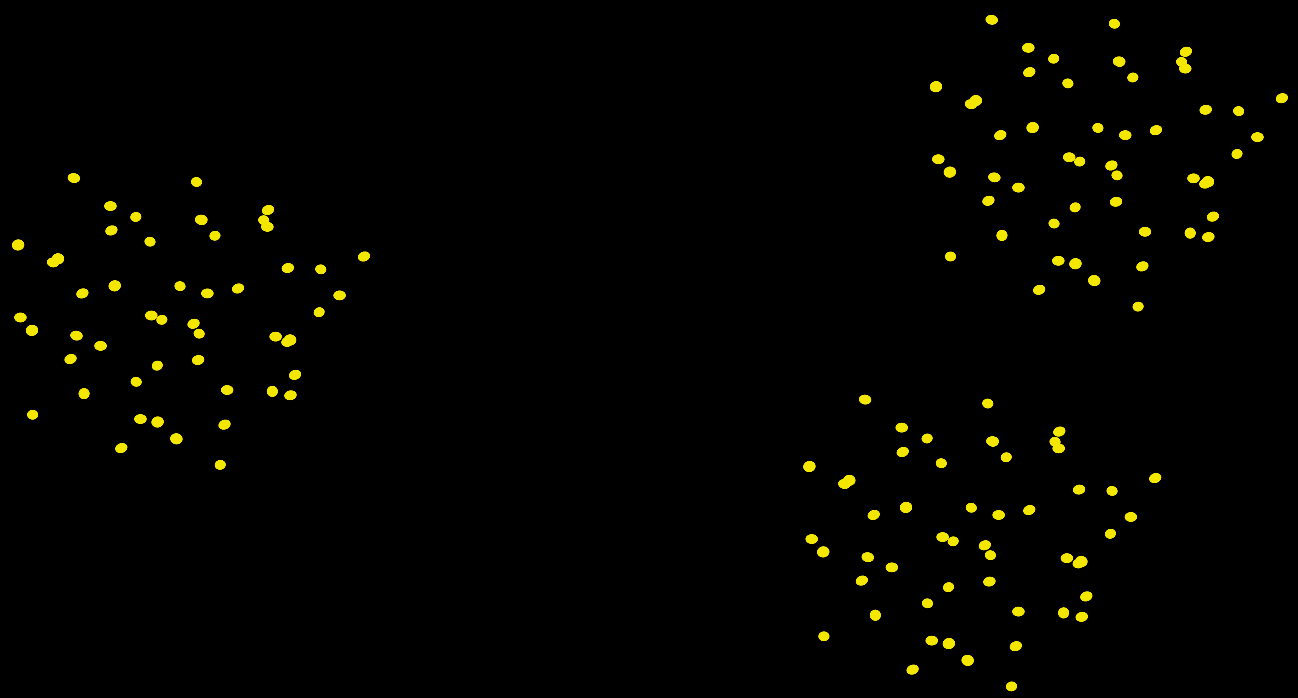
159 KB

Clustered, k=4

53 KB

Issue 1: Local Optima

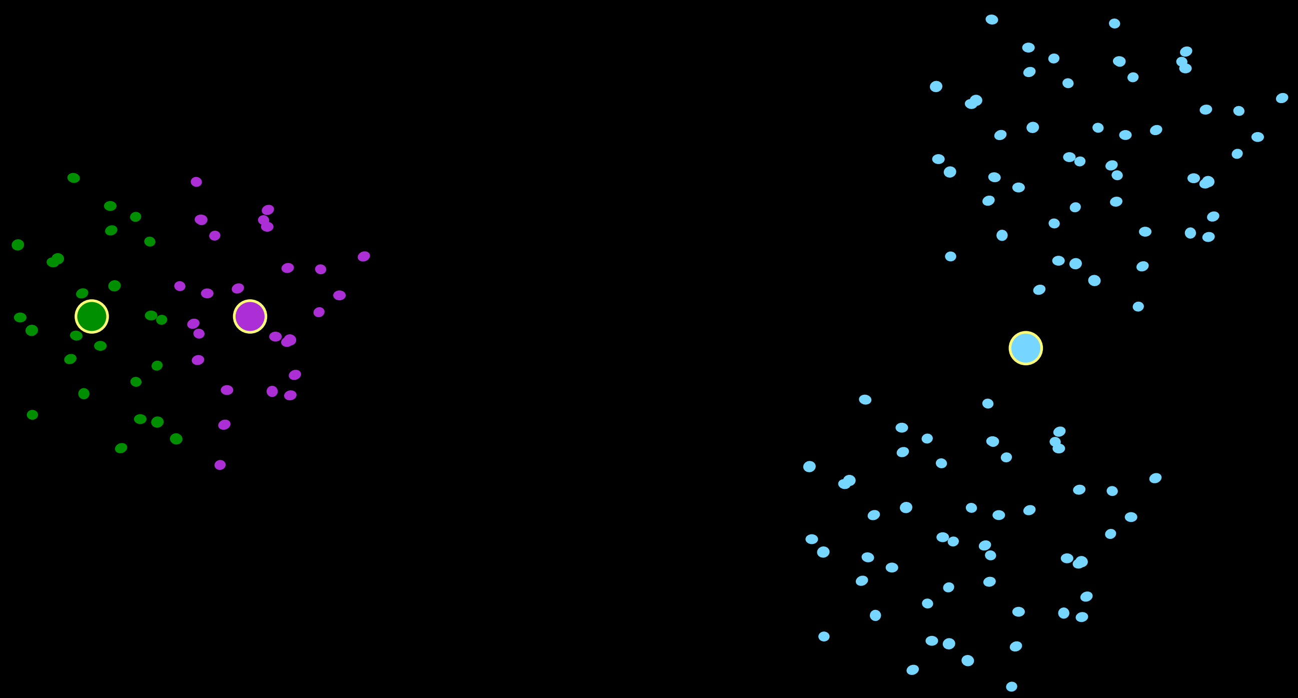
- K-means is greedy!
- Converging to a non-global optimum:



[Example from Andrew Moore]

Issue 1: Local Optima

- K-means is greedy!
- Converging to a non-global optimum:



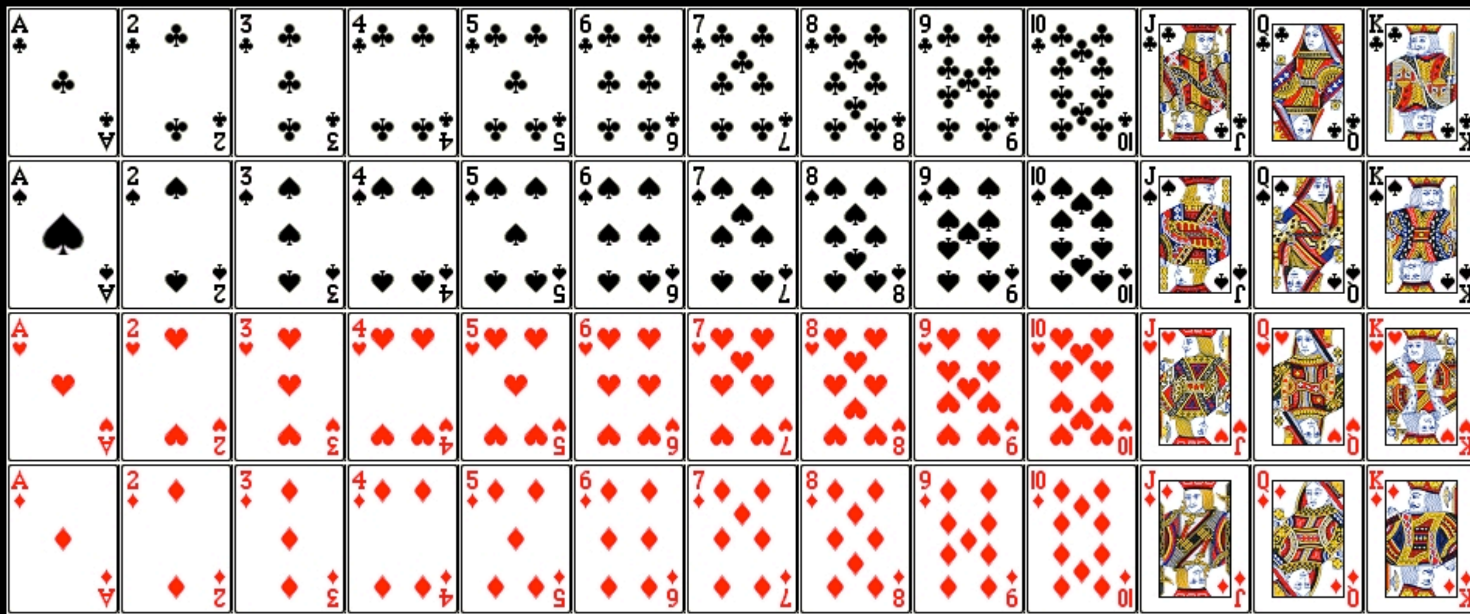
[Example from Andrew Moore]

Issue 2: How long will it take?

- We don't know!
- K-means is O(nkdI)
  - d = # features (dimensionality)
  - I =# iterations
- # iterations depends on random initialization
  - "Good" init: few iterations
  - "Bad" init: lots of iterations
  - How can we tell the difference, before clustering?
    - We can't
    - Use heuristics to guess "good" init

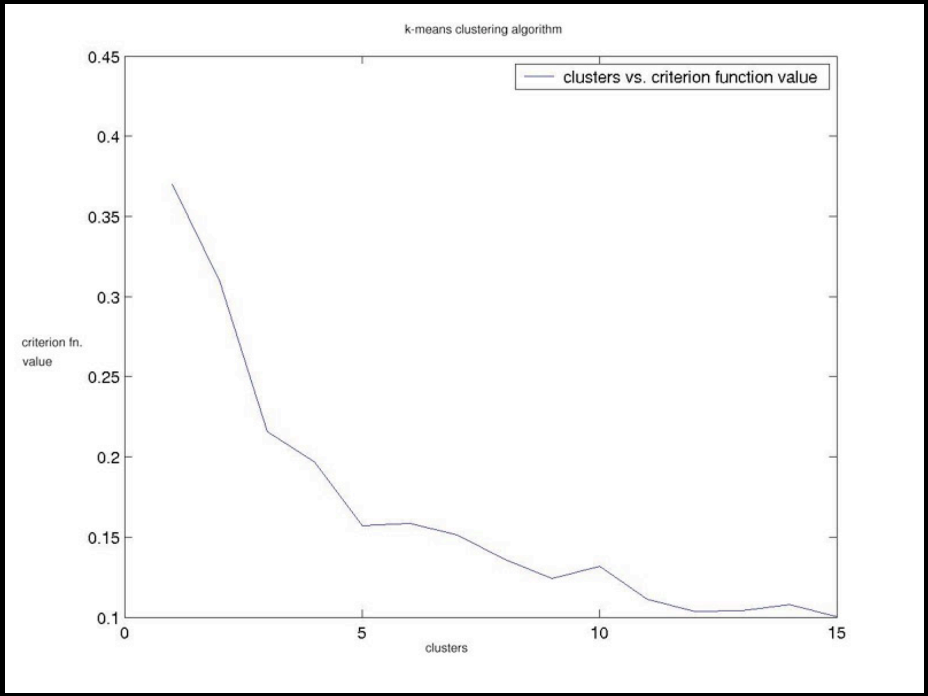
Issue 3: How many clusters?

- The "Holy Grail" of clustering



Issue 3: How many clusters?

- Select *k* that gives partition with least variance?

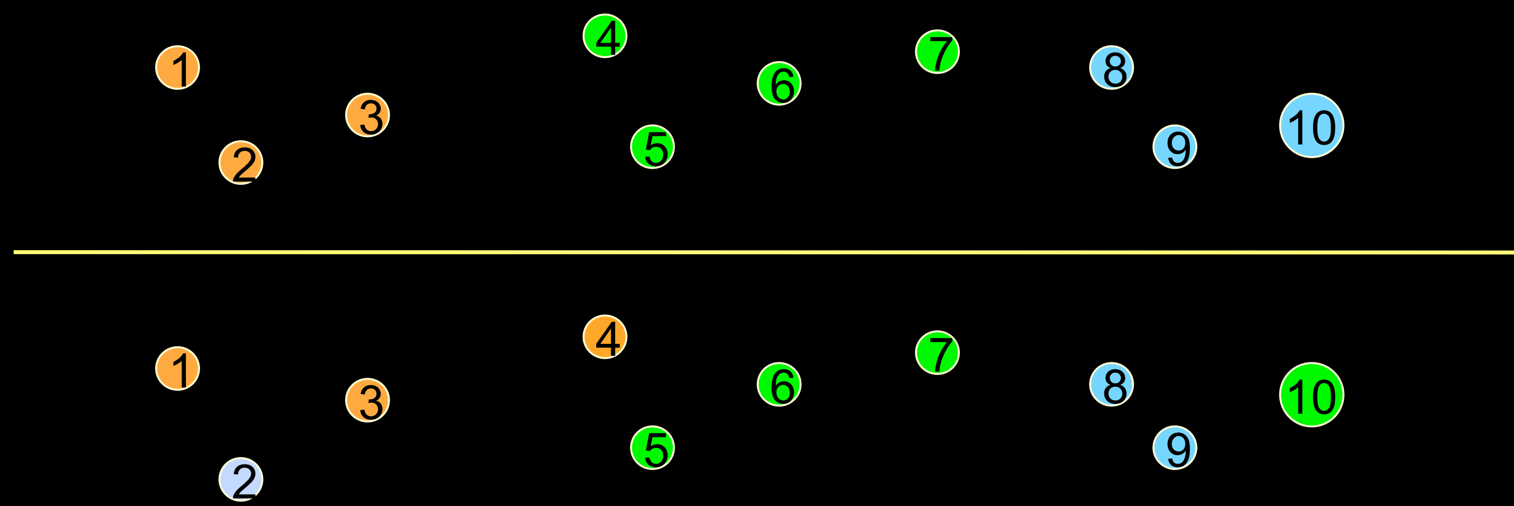


[Dhande and Fiore, 2002]

- Best *k* depends on the user's goal

Issue 4: How good is the result?

- Rand Index
  - A = # pairs in same cluster in both partitions
  - B = # pairs in different clusters in both partitions
  - Rand = (A + B) / Total number of pairs



Rand = (5 + 26) / 45

K-means: Parametric or Non-parametric?

- Cluster models: means
- Data models?

- All clusters are spherical
  - Distance in any direction is the same
  - Cluster may be arbitrarily "big" to include outliers

EM Clustering

- Parametric solution
  - Model the data distribution
- Each cluster: Gaussian model $\mathcal{N}(\mu,\sigma)$
  - Data: "mixture of models"
- Hidden value z^t is the cluster of item t
- E-step: estimate cluster memberships

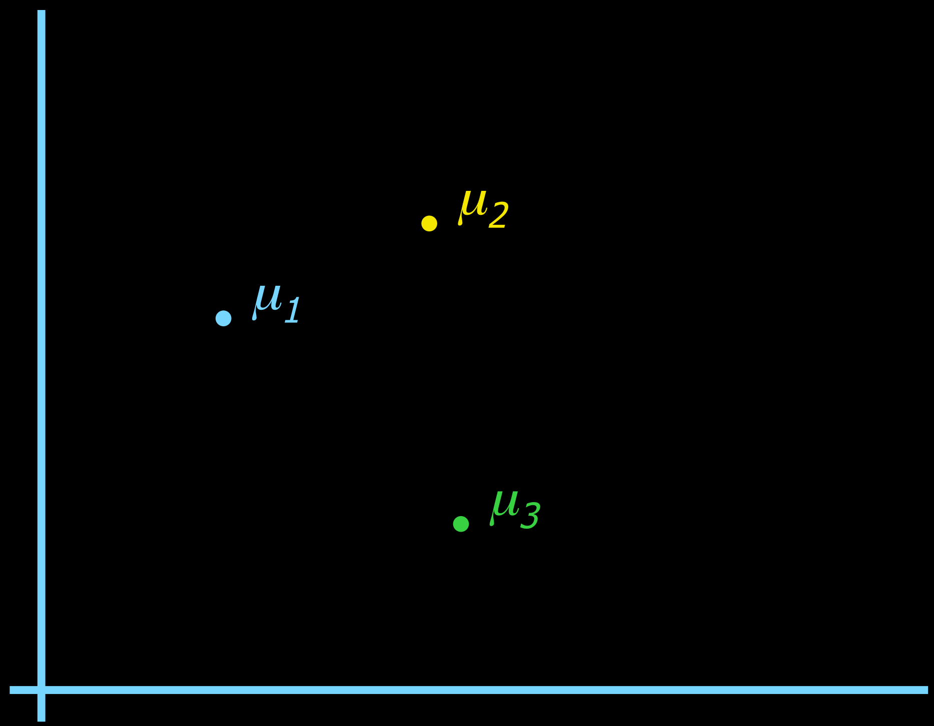
$$E\left[z^t \mid X,\mu,\sigma\right] = \frac{p\left(\mathbf{x}^t \mid C,\mu,\sigma\right)P(C)}{\sum_j p\left(\mathbf{x}^t \mid C_j,\mu_j,\sigma_j\right)P\left(C_j\right)}$$

- M-step: maximize likelihood (clusters, params)

$$\mathcal{L}\left(\mu,\sigma \mid X\right) = P(X \mid \mu,\sigma)$$

The GMM assumption

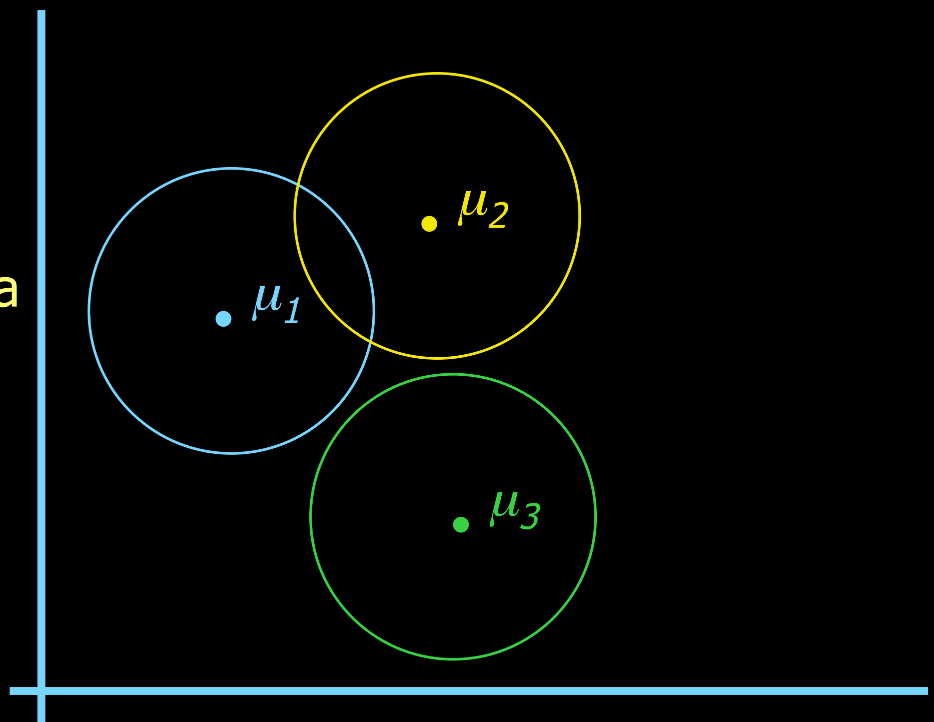
- There are k components. The i'th component is called ω_i
- Component ω_i has an associated mean vector μ_i



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 $\sigma^2\mathbf{I}$

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

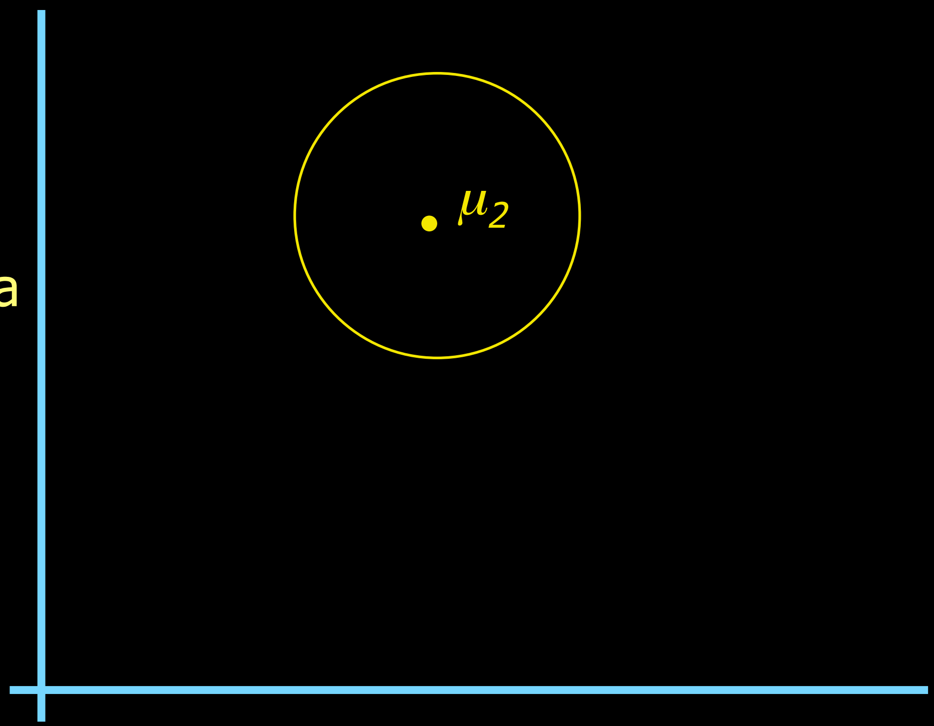


The GMM assumption

- 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 $P(\omega_i)$.

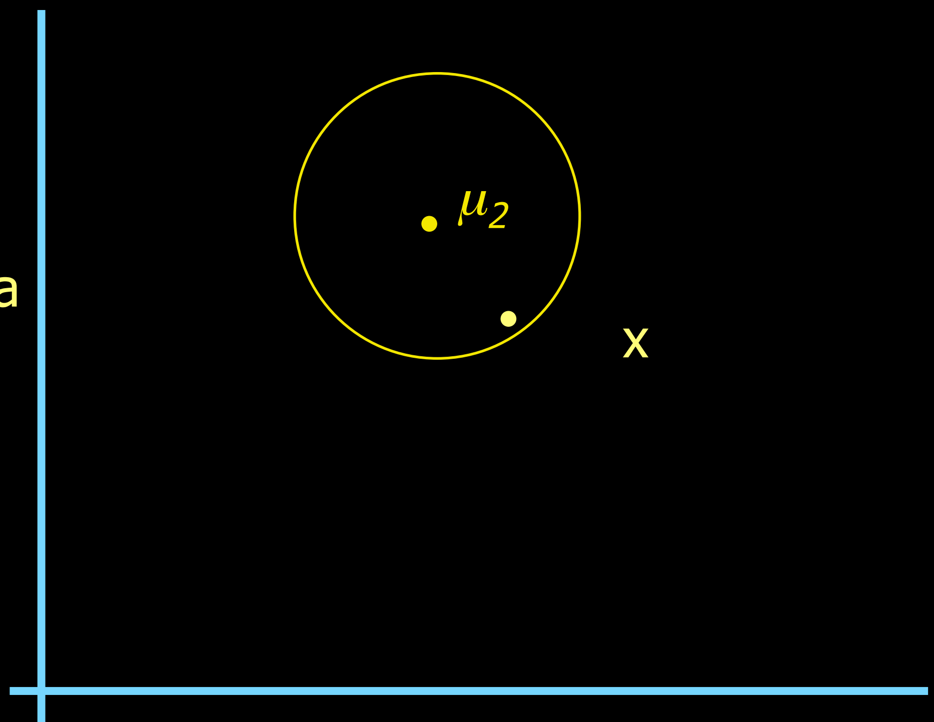


The GMM assumption

- 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 $\sigma^2 \boldsymbol{I}$

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

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

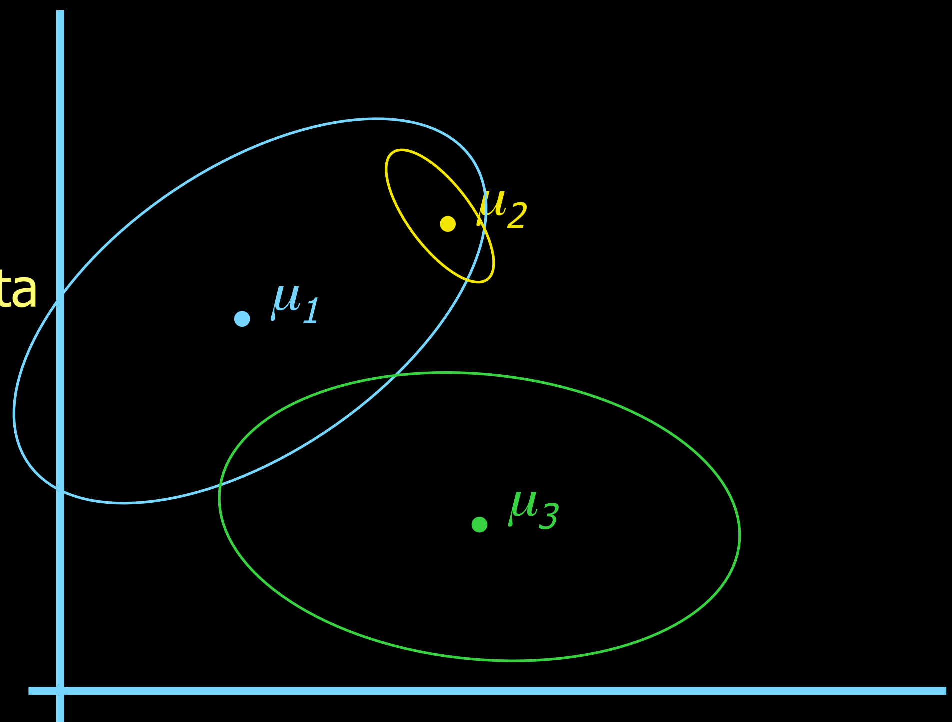


The General GMM assumption

- 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:

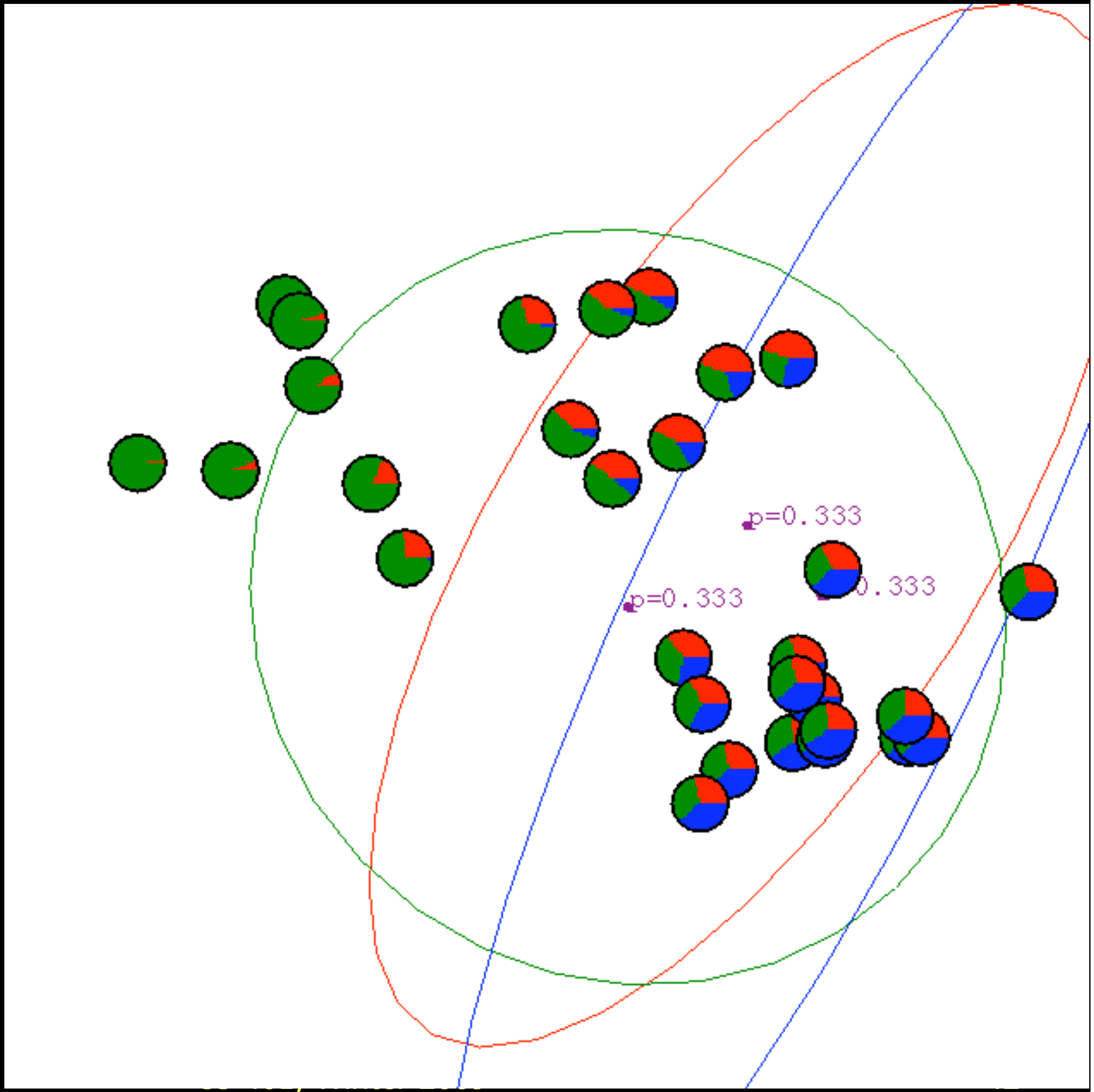
1. Pick a component at random. Choose component i with probability $P(\omega_i)$.
2. Datapoint ~ $N(\mu_i, \Sigma_i)$



EM in action

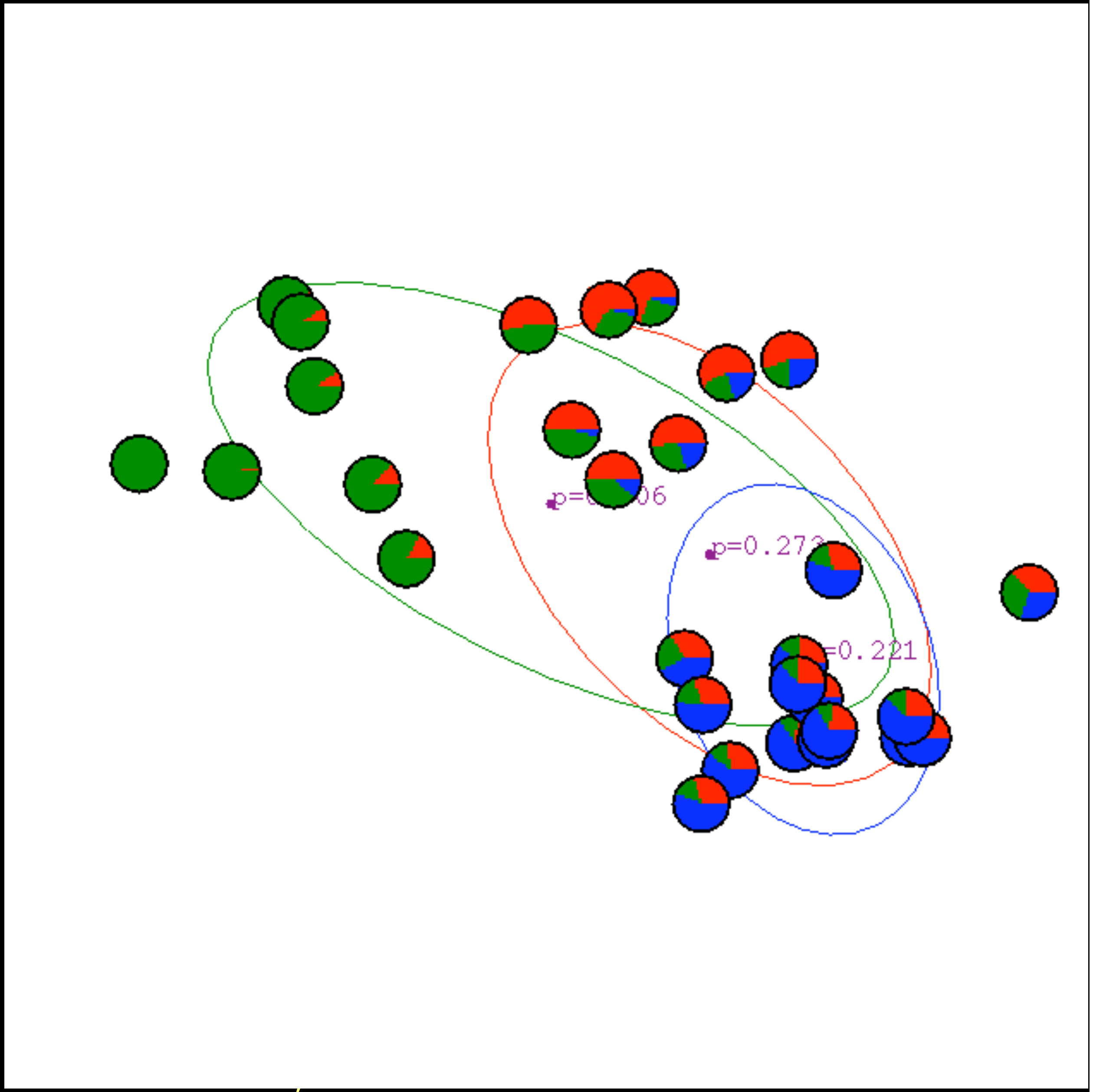
- http://www.the-wabe.com/notebook/em-algorithm.html

Gaussian Mixture Example: Start



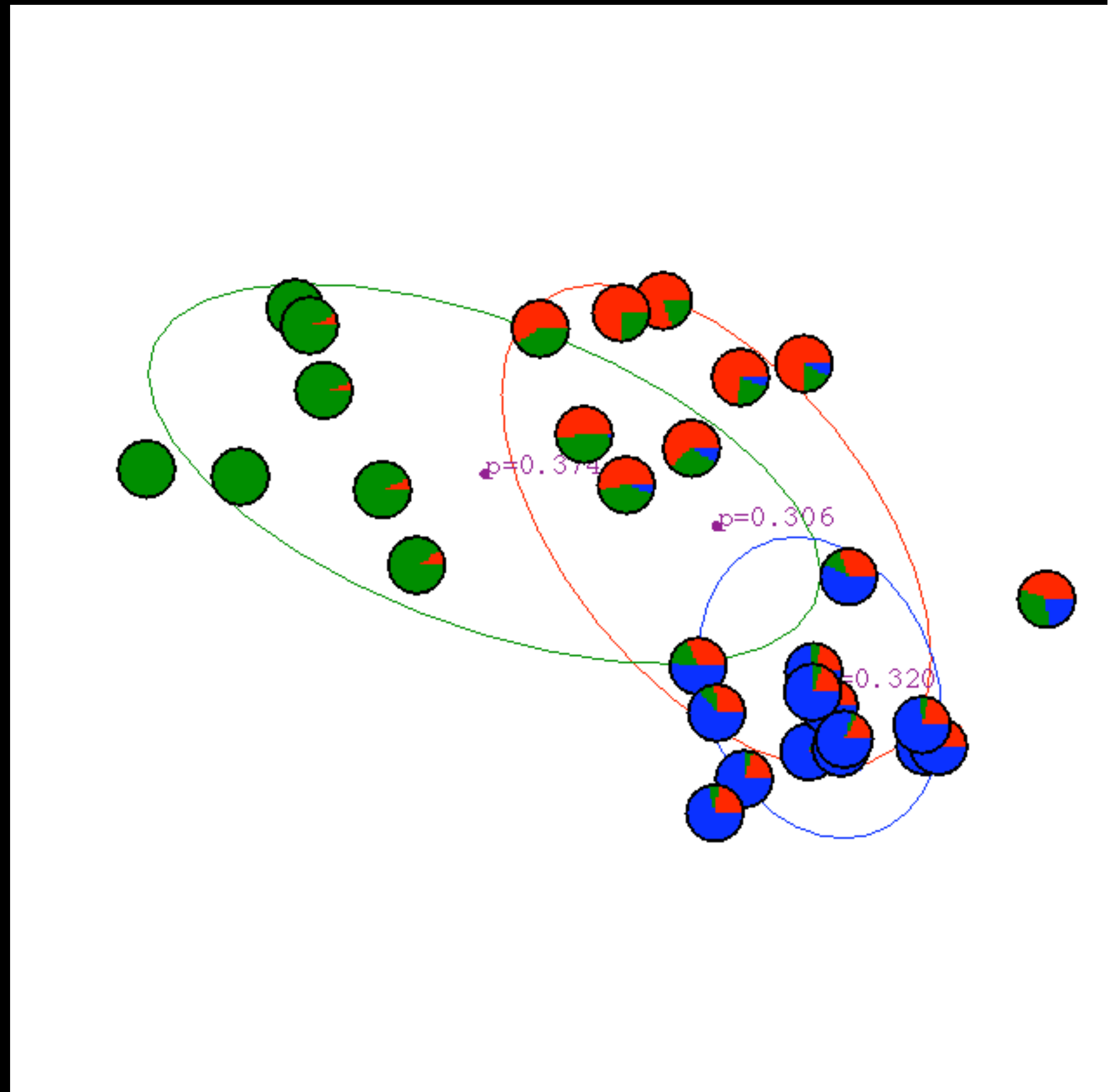
[© Andrew Moore]

After first iteration



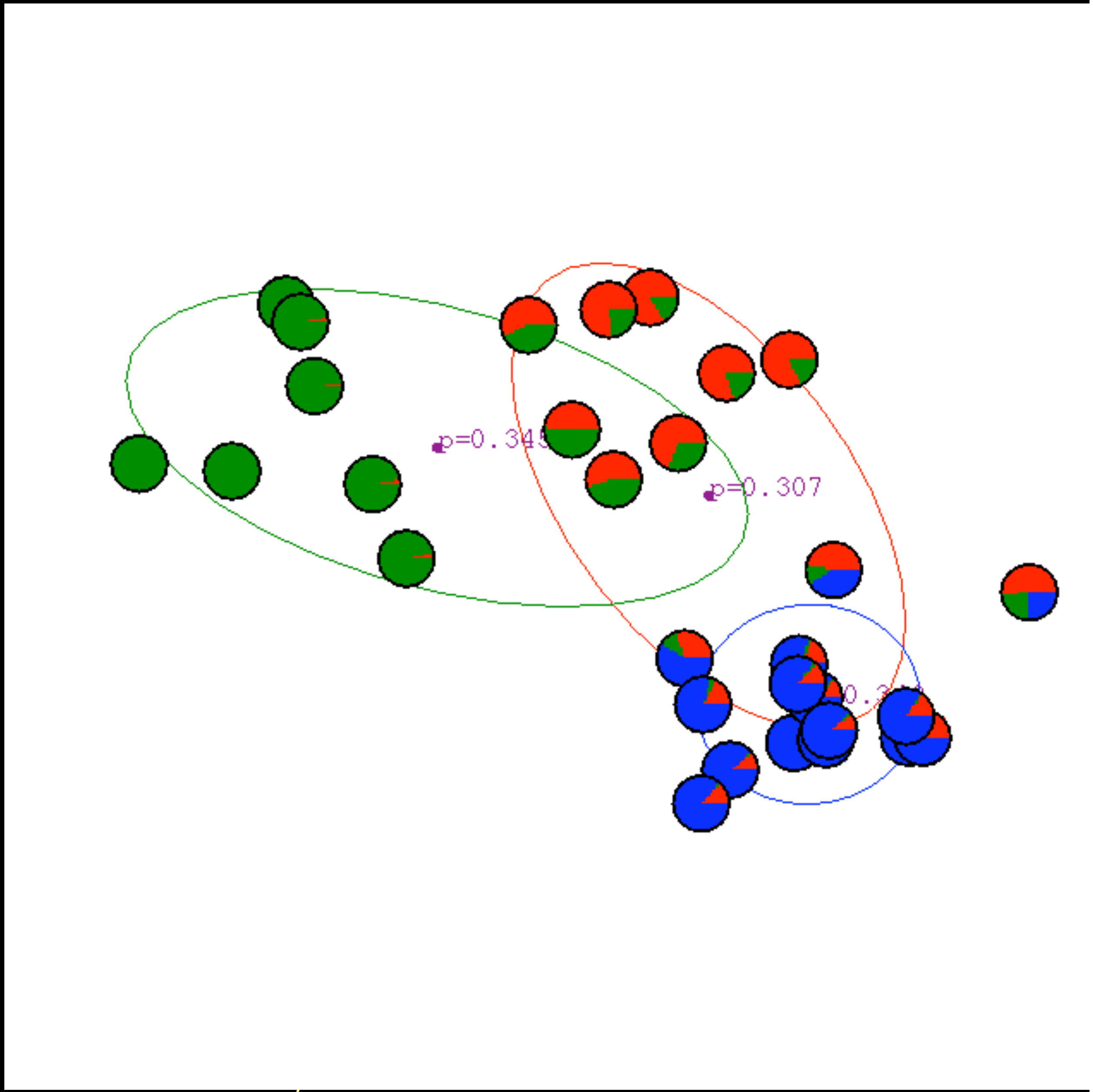
[© Andrew Moore]

After 2nd iteration



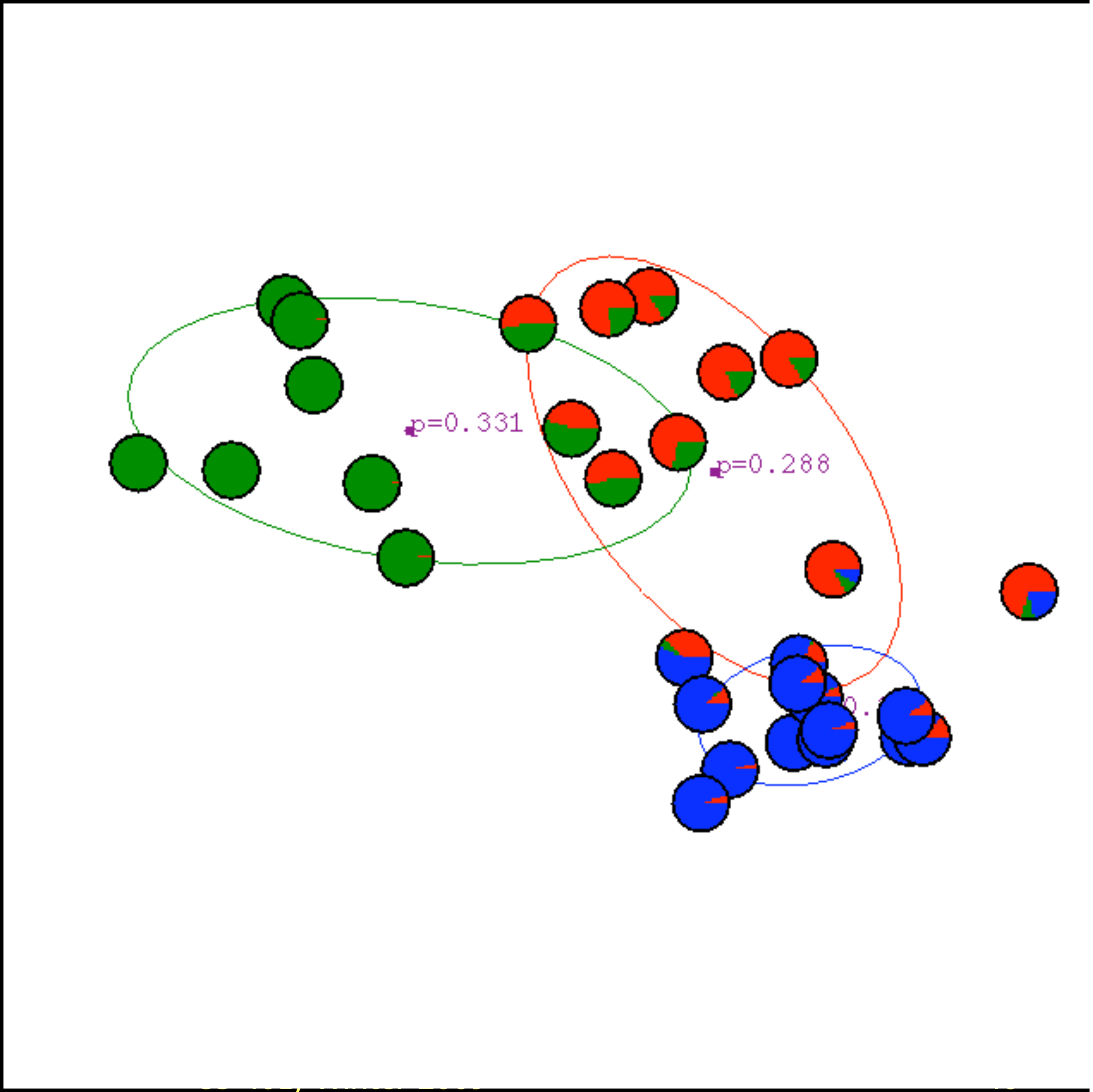
[© Andrew Moore]

After 3rd iteration



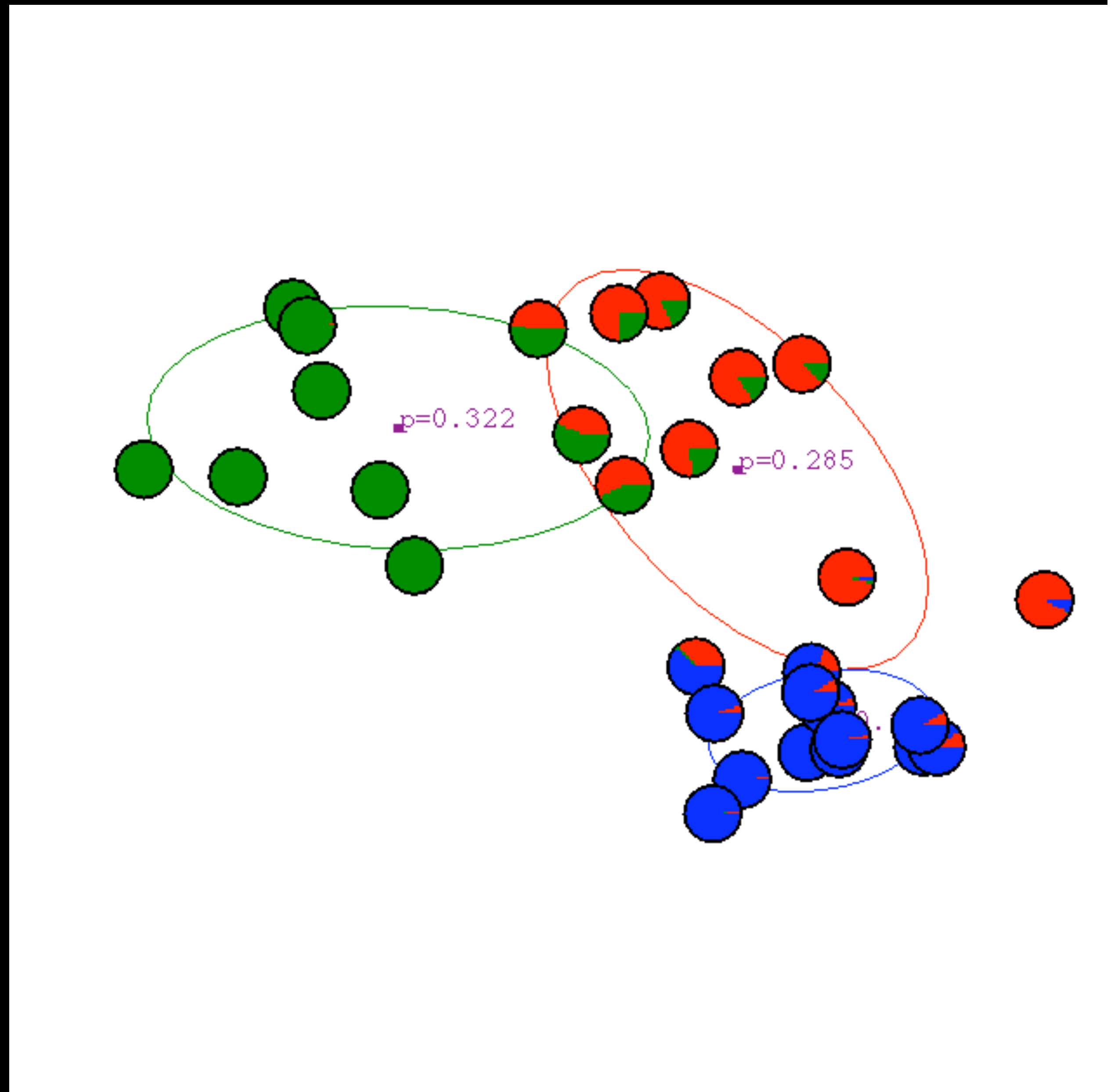
[© Andrew Moore]

After 4th iteration



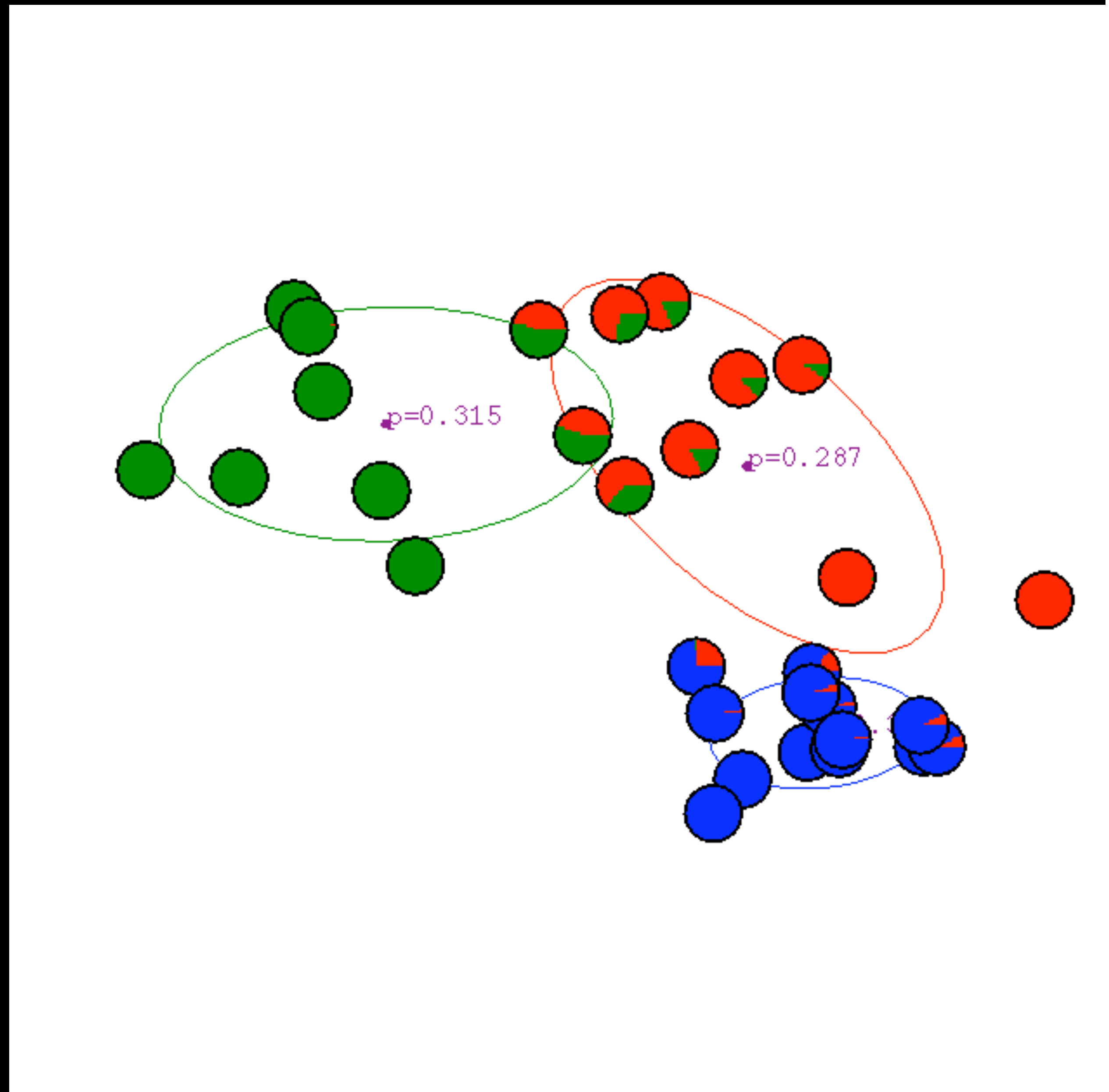
[© Andrew Moore]

After 5th iteration



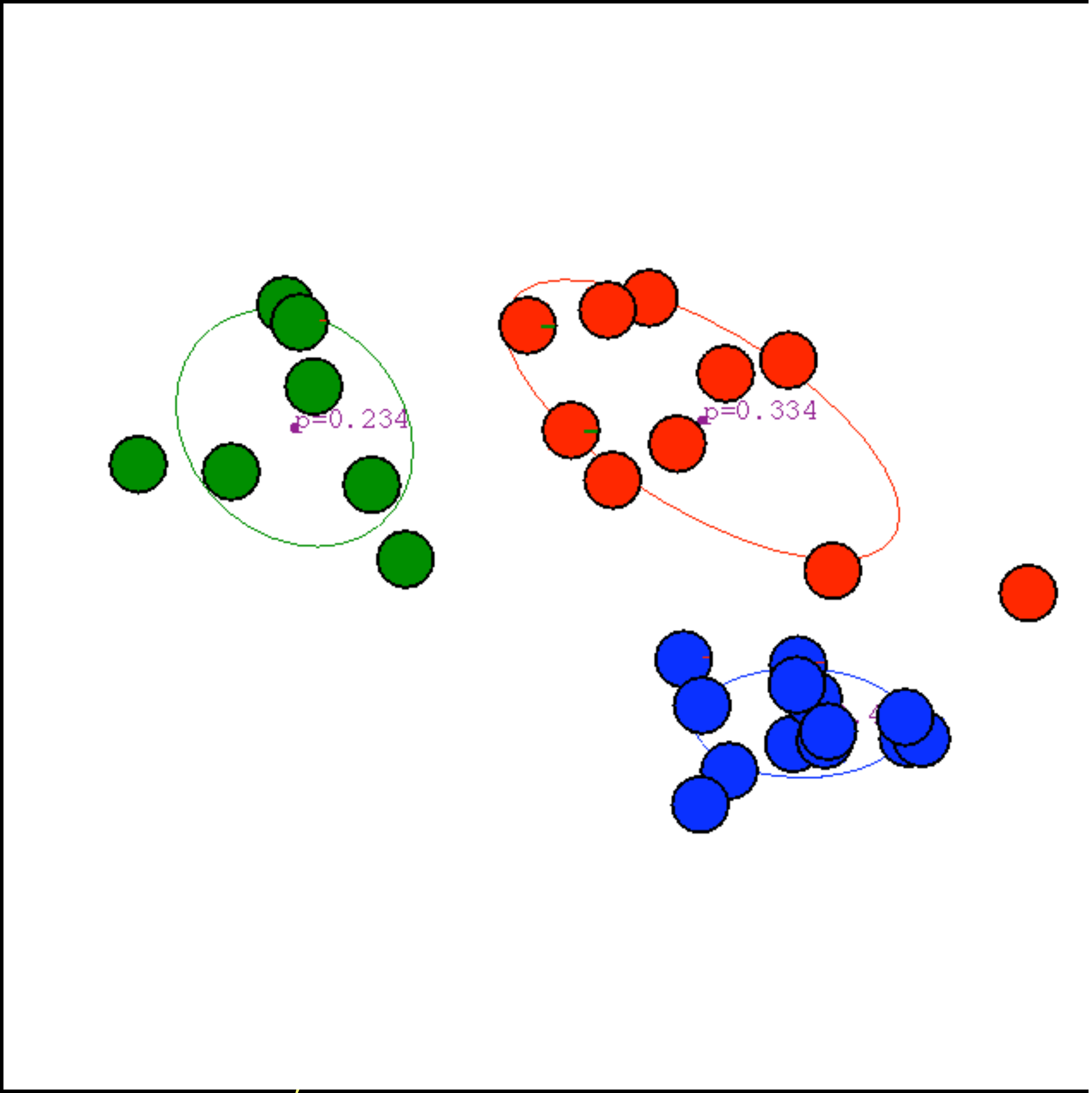
[© Andrew Moore]

After 6th iteration



[© Andrew Moore]

After 20th iteration



[© Andrew Moore]

EM Benefits

- Model actual data distribution, not just centers
- Get probability of membership in each cluster, not just distance
- Clusters do not need to be “round”

EM Issues?

- Local optima
- How long will it take?
- How many clusters?
- Evaluation

Summary: Key Points for Today

- Unsupervised Learning
  - Why? How?
- K-means Clustering
  - Iterative
  - Sensitive to initialization
  - Non-parametric
  - Local optimum
  - Rand Index
- EM Clustering
  - Iterative
  - Sensitive to initialization
  - Parametric
  - Local optimum

Next Time

- Clustering Reading: Alpaydin Ch. 7.1-7.4, 7.8
- Reading questions: Gavin, Ronald, Matthew

- Next time: Reinforcement learning – Robots!