THE UNIVERSITY of TENNESSEE KNOXVILLE

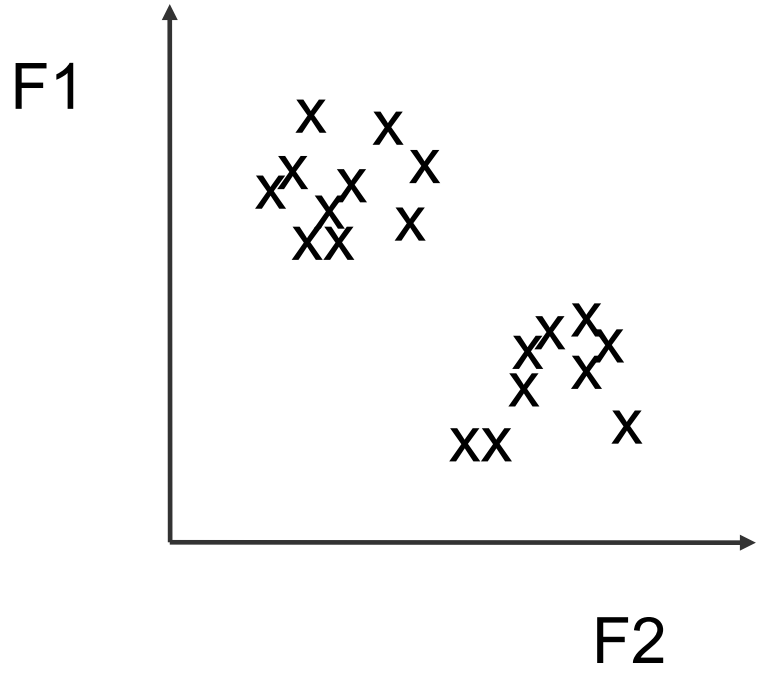
Lecture 12: Clustering

Instructor: Sergei V. Kalinin

Clustering

- The process of grouping a set of objects into classes of similar objects
  - Objects within a cluster should be similar.
  - Objects from different clusters should be dissimilar.
- The most common form of *unsupervised learning*

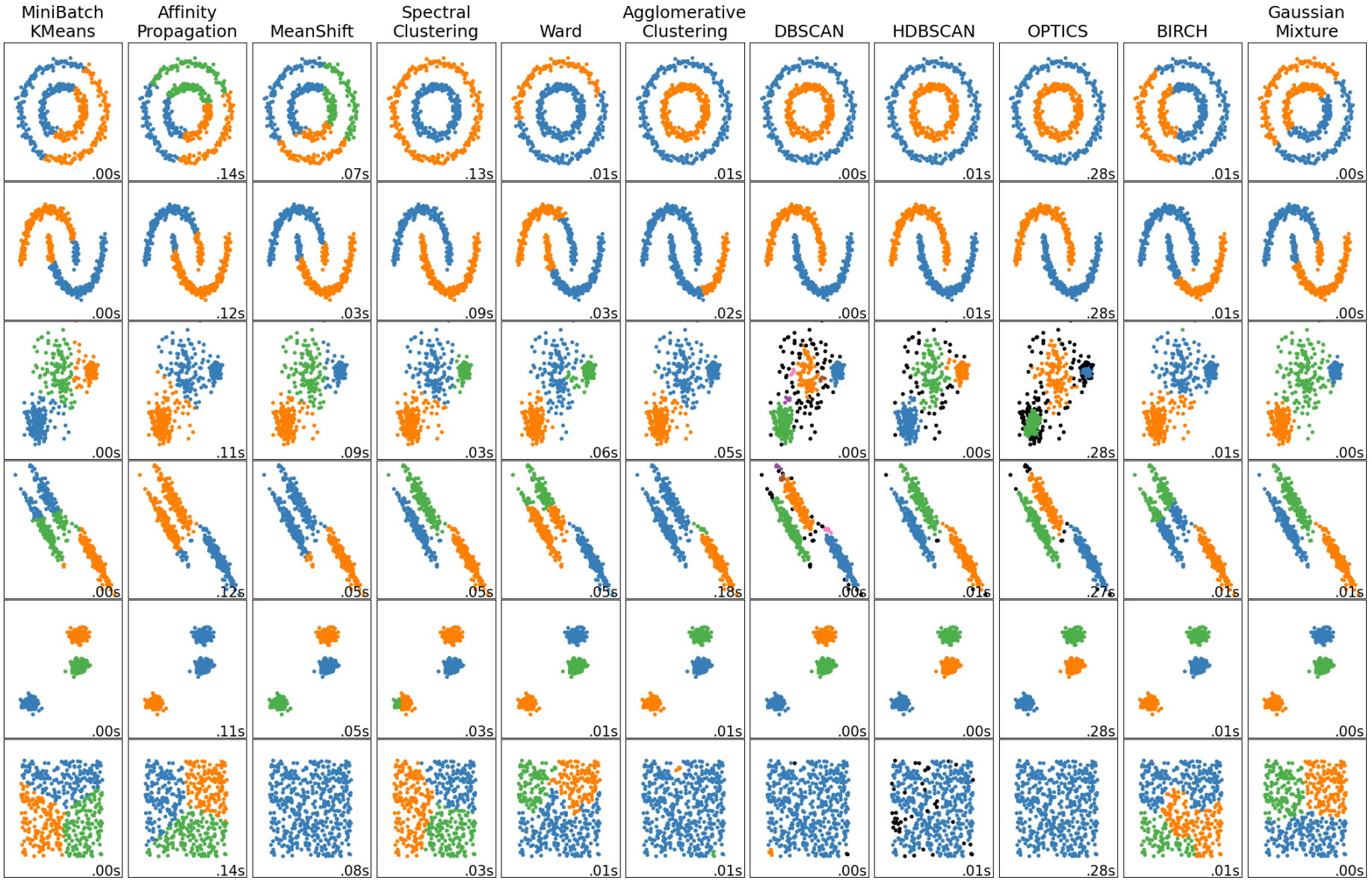
- Given a set of data points, each described by a set of attributes, find clusters such that:
  - Inter-cluster similarity is maximized
  - Intra-cluster similarity is minimized
- Requires the definition of a similarity measure



https://en.wikipedia.org/wiki/Cluster_analysis

CS276: Information Retrieval and Web Search Pandu Nayak and Prabhakar Raghavan

Clustering: good news and bad news



https://scikit-learn.org/stable/auto_examples/cluster/plot_cluster_comparison.html

Clustering: Data Structures

- Hierarchical clustering
- K-means clustering
- Gaussian Mixture Models
- Density-based clustering
- Spectral clustering

Data matrix (two modes)

$$\begin{bmatrix} x_{11} & \cdots & x_{1f} & \cdots & x_{1p} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{i1} & \cdots & x_{if} & \cdots & x_{ip} \\ \cdots & \cdots & \cdots & \cdots & \cdots \\ x_{n1} & \cdots & x_{nf} & \cdots & x_{np} \end{bmatrix}$$

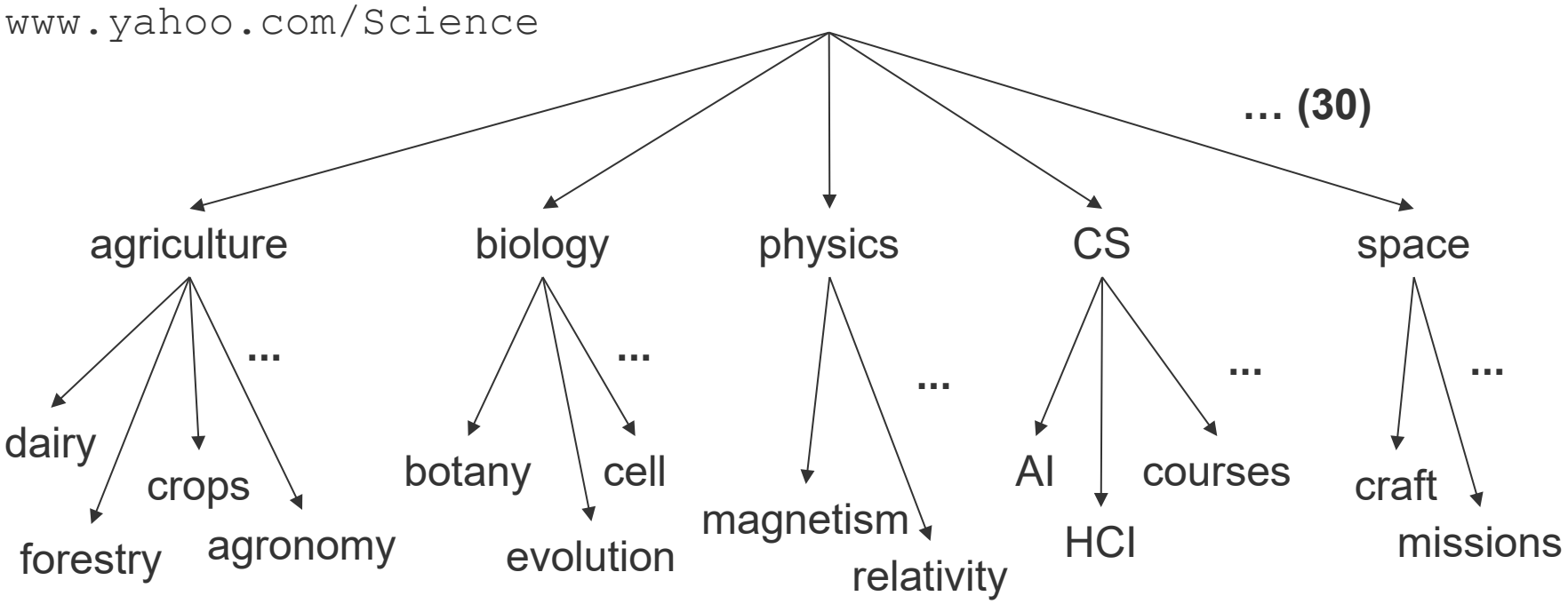
Dissimilarity matrix (one mode)

$$\begin{bmatrix} 0 & & & & \\ d(2,1) & 0 & & & \\ d(3,1) & d(3,2) & 0 & & \\ \vdots & \vdots & \vdots & & \\ d(n,1) & d(n,2) & \cdots & \cdots & 0 \end{bmatrix}$$

From: Han and Kamber, Data Mining: Concepts and Techniques

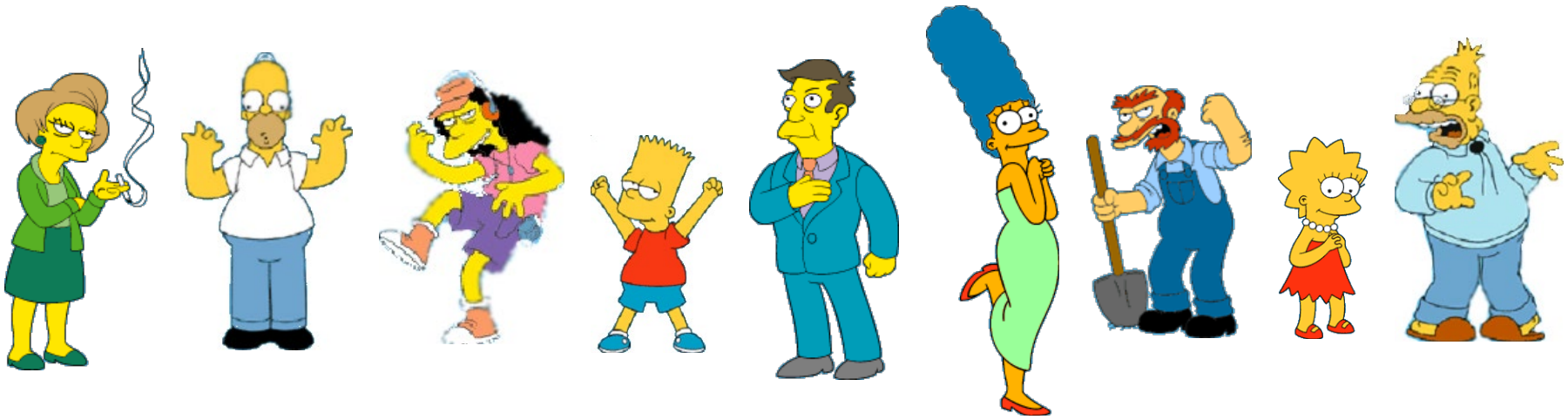
Clustering

Yahoo! Hierarchy *isn't* clustering but *is* the kind of output you want from clustering



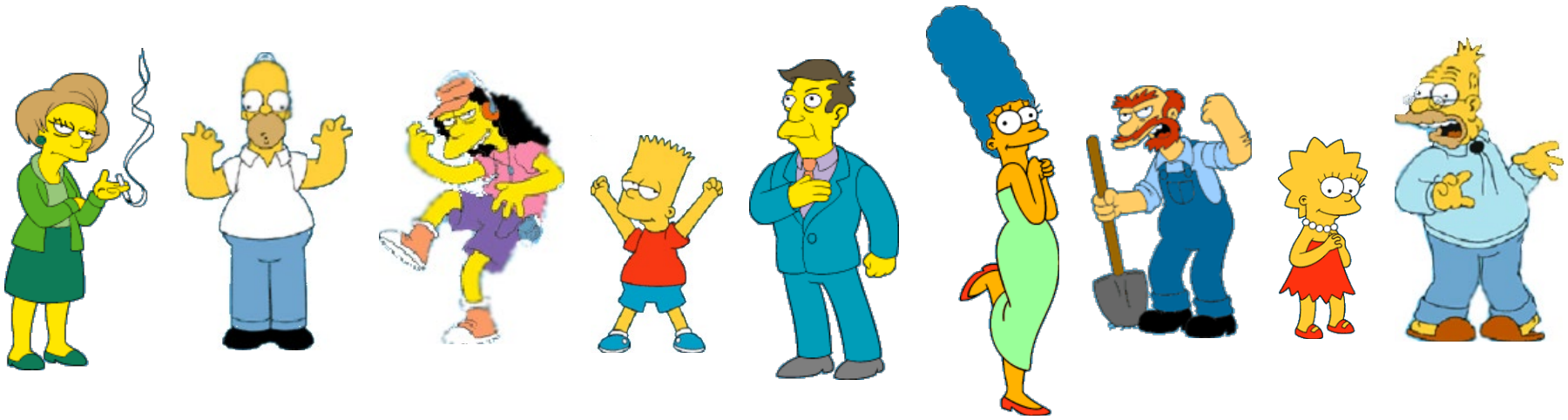
What is a natural grouping of these objects?

Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

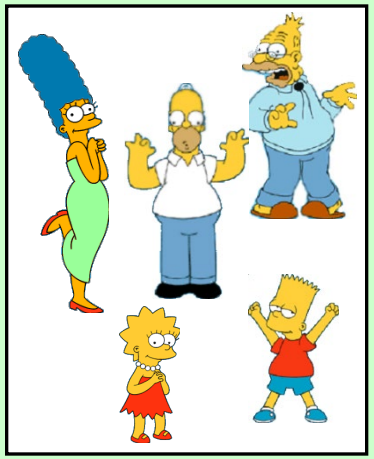


What is a natural grouping of these objects?

Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

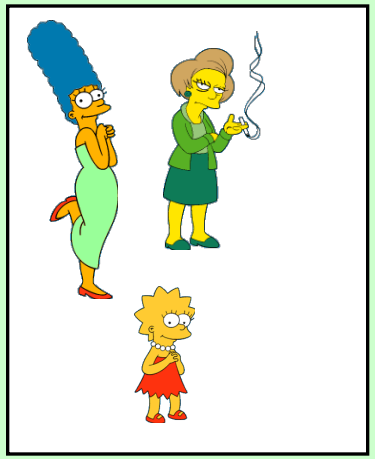


Clustering is subjective



Simpson's Family

School Employees



Females

Males

What is Similarity?

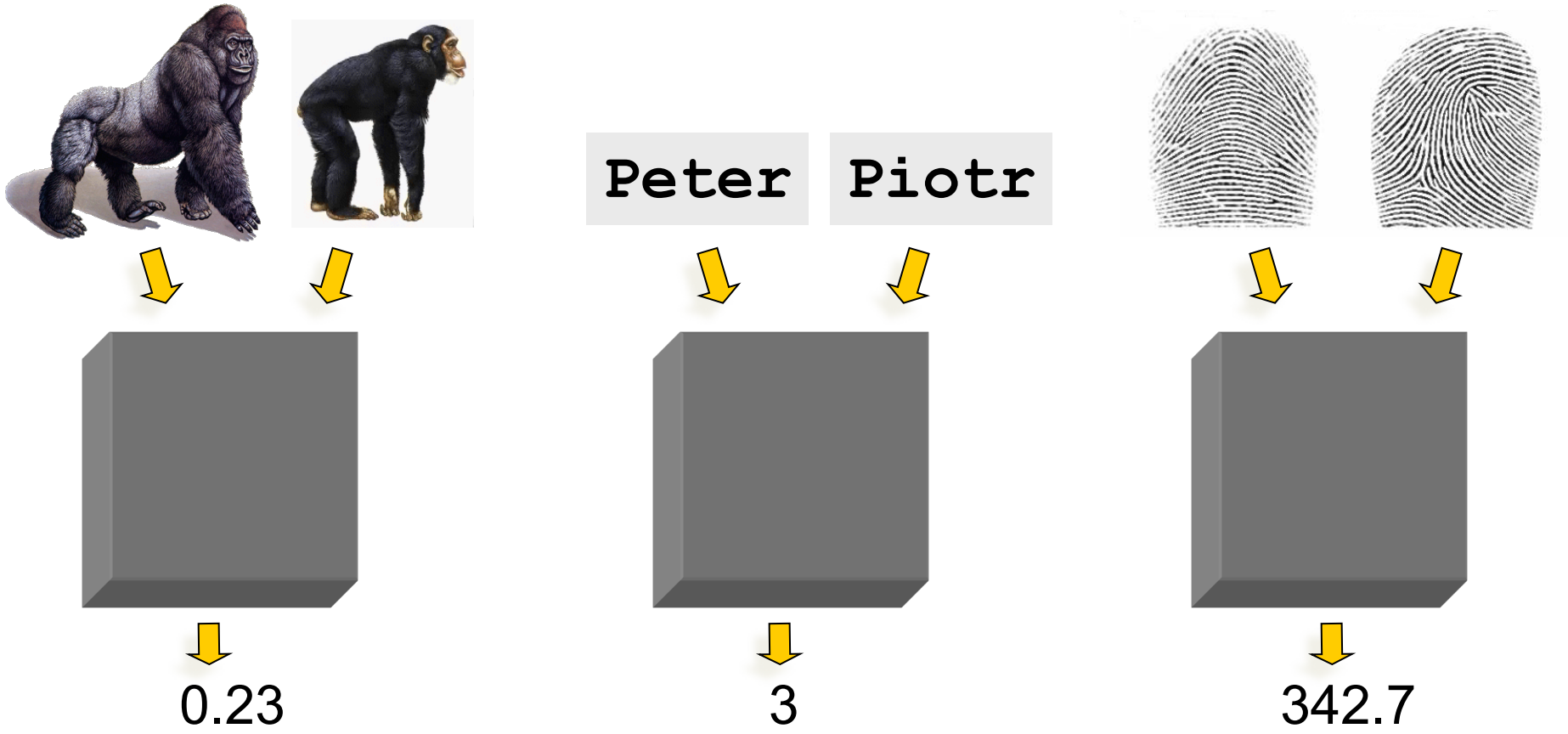
Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

Similarity is hard to define, but…
"*We know it when we see it*"

Defining Distance Measures

Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

Definition: Let O_1 and O_2 be two objects from the universe of possible objects. The distance (dissimilarity) between O_1 and O_2 is a real number denoted by $D(O_1,O_2)$



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

What properties should a distance measure have?

- $D(A,B) = D(B,A)$ *Symmetry*
- $D(A,A) = 0$ *Constancy of Self-Similarity*
- $D(A,B) = 0$ iif $A = B$ *Positivity (Separation)*
- $D(A,B) \leq D(A,C) + D(B,C)$ *Triangular Inequality*

Intuitions behind desirable distance measure properties

$D(A,B) = D(B,A)$

Otherwise you could claim "Alex looks like Bob, but Bob looks nothing like Alex."

$D(A,A) = 0$

Otherwise you could claim "Alex looks more like Bob, than Bob does."

Intuitions behind desirable distance measure properties (continued)

$D(A,B) = 0$ IIf A=B

Otherwise there are objects in your world that are different, but you cannot tell apart.

$D(A,B) \leq D(A,C) + D(B,C)$

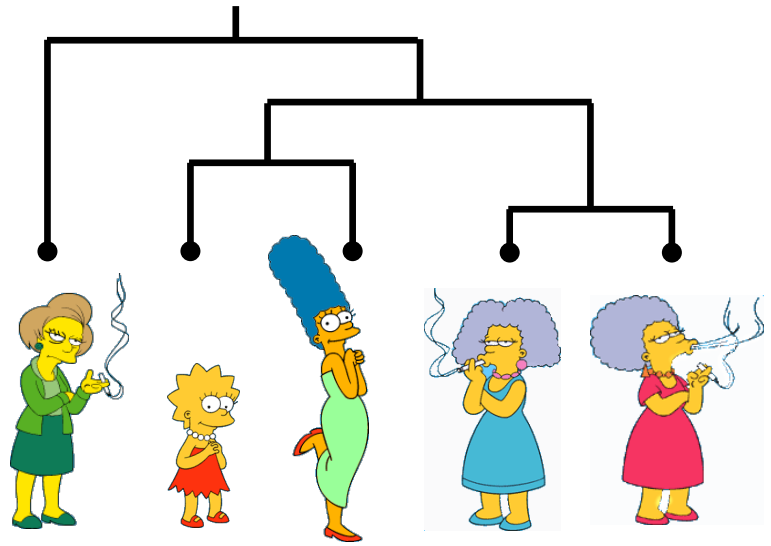
Otherwise you could claim "Alex is very like Bob, and Alex is very like Carl, but Bob is very unlike Carl."

Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

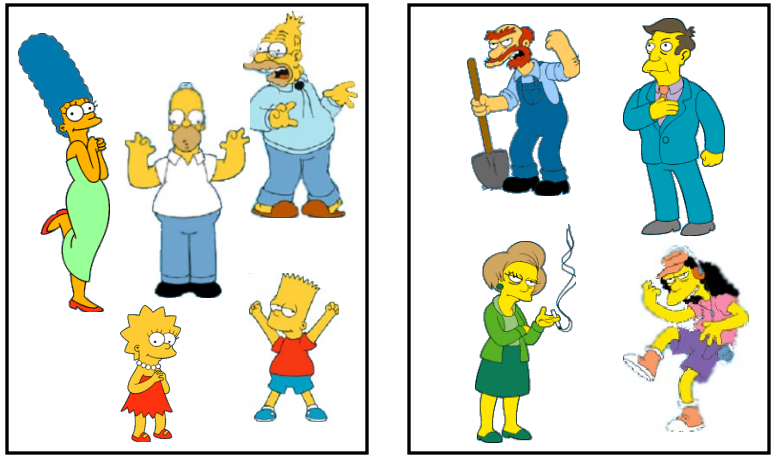
Two Types of Clustering

- **Partitional algorithms:** Construct various partitions and then evaluate them by some criterion
- **Hierarchical algorithms:** Create a hierarchical decomposition of the set of objects using some criterion

Hierarchical

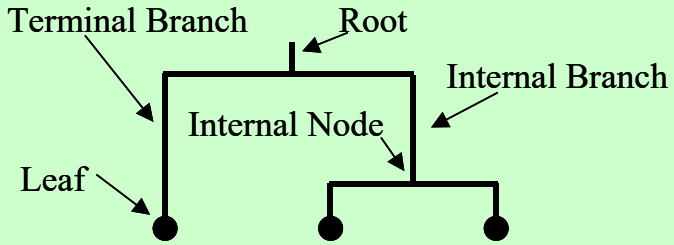


Partitional

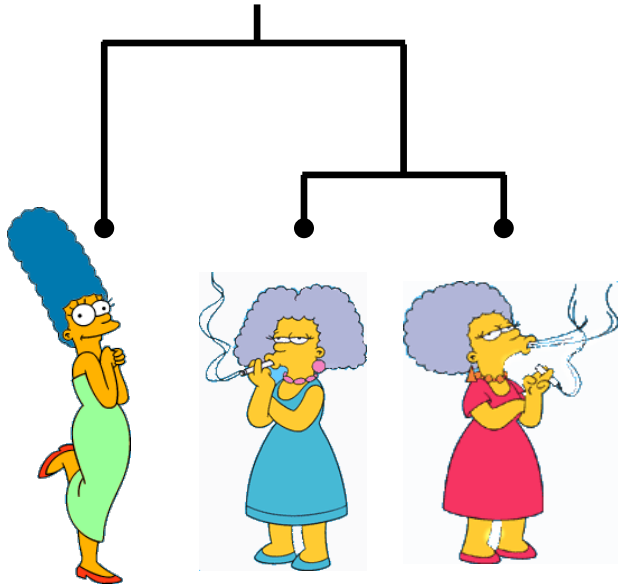


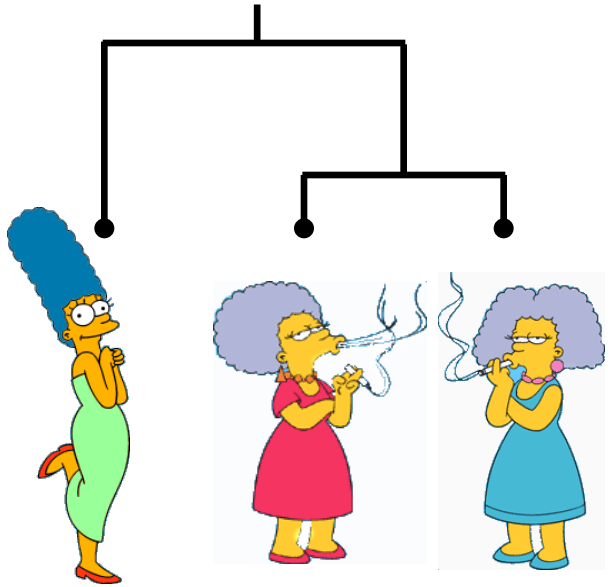
Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

Dendogram: A Useful Tool for Summarizing Similarity Measurements



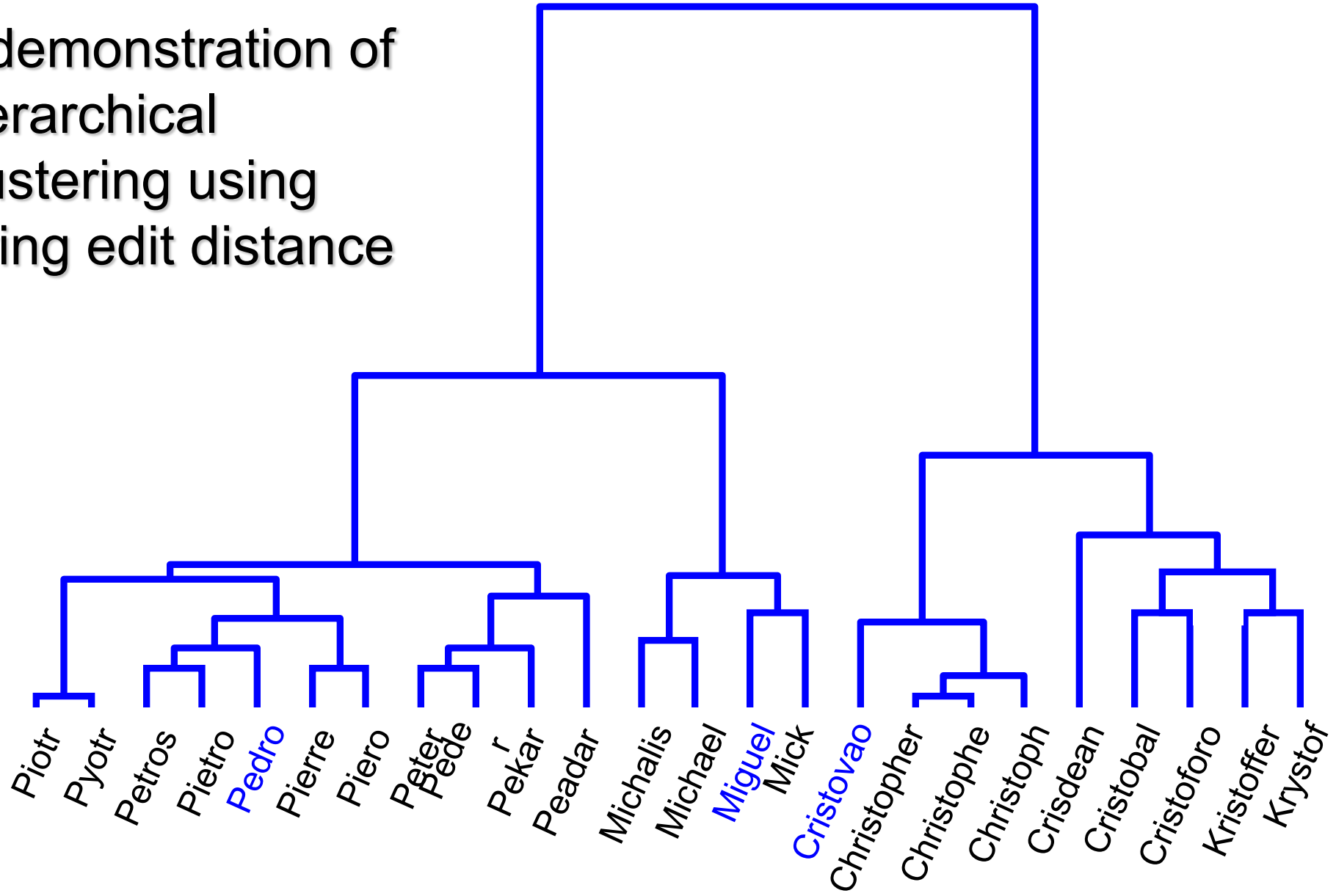
The similarity between two objects in a dendrogram is represented as the height of the lowest internal node they share.





Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

A demonstration of hierarchical clustering using string edit distance

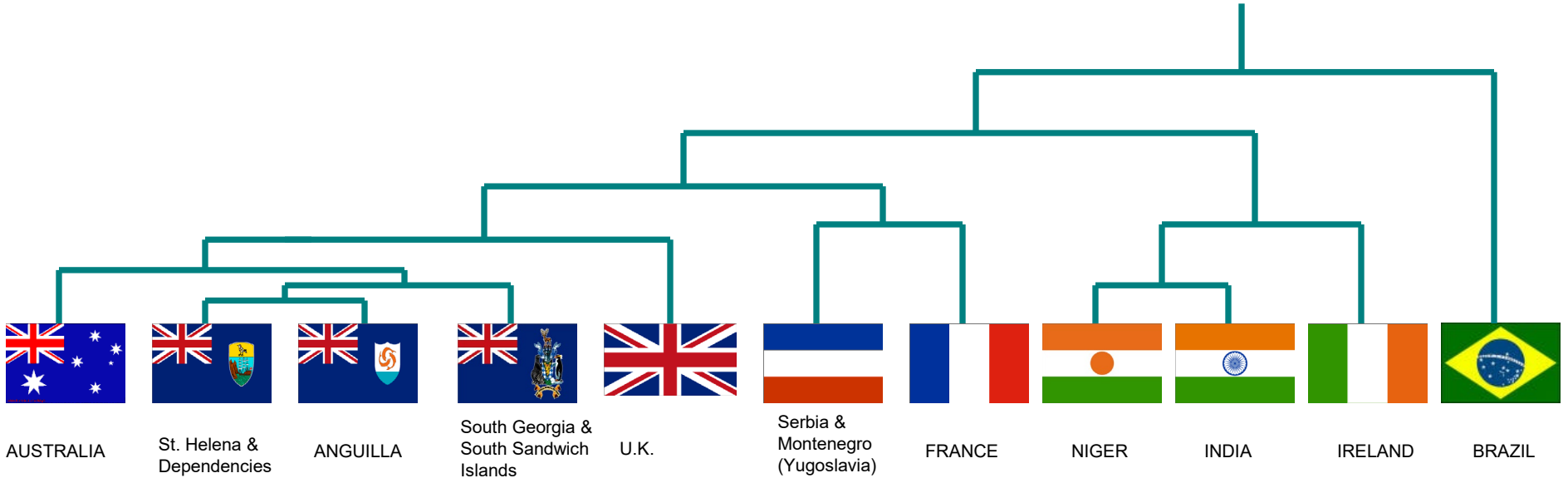


Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

Hierarchal clustering can sometimes show patterns that are meaningless or spurious

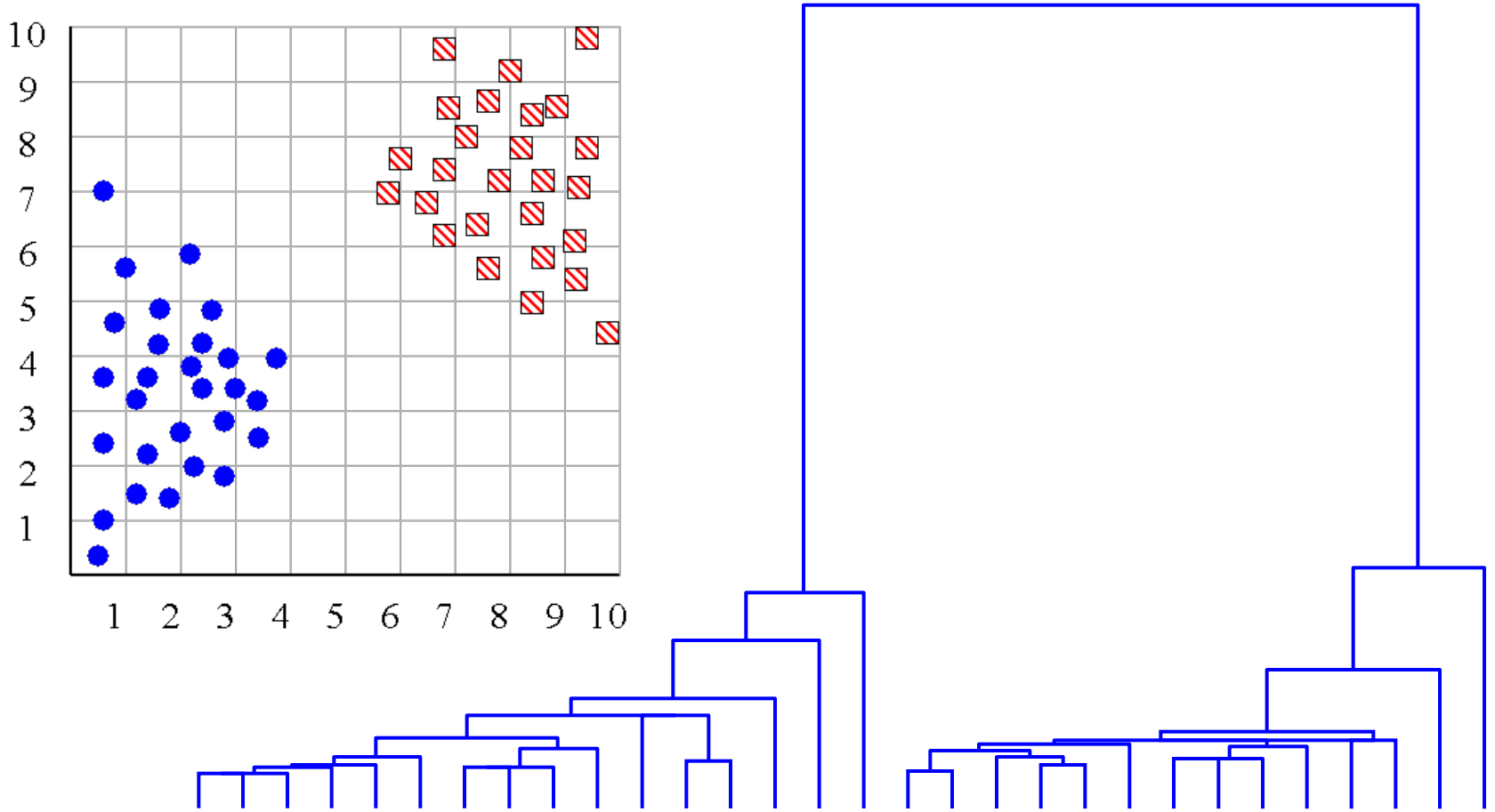
The tight grouping of Australia, Anguilla, St. Helena etc is meaningful; all these countries are former UK colonies

However the tight grouping of Niger and India is completely spurious; there is no connection between the two.



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

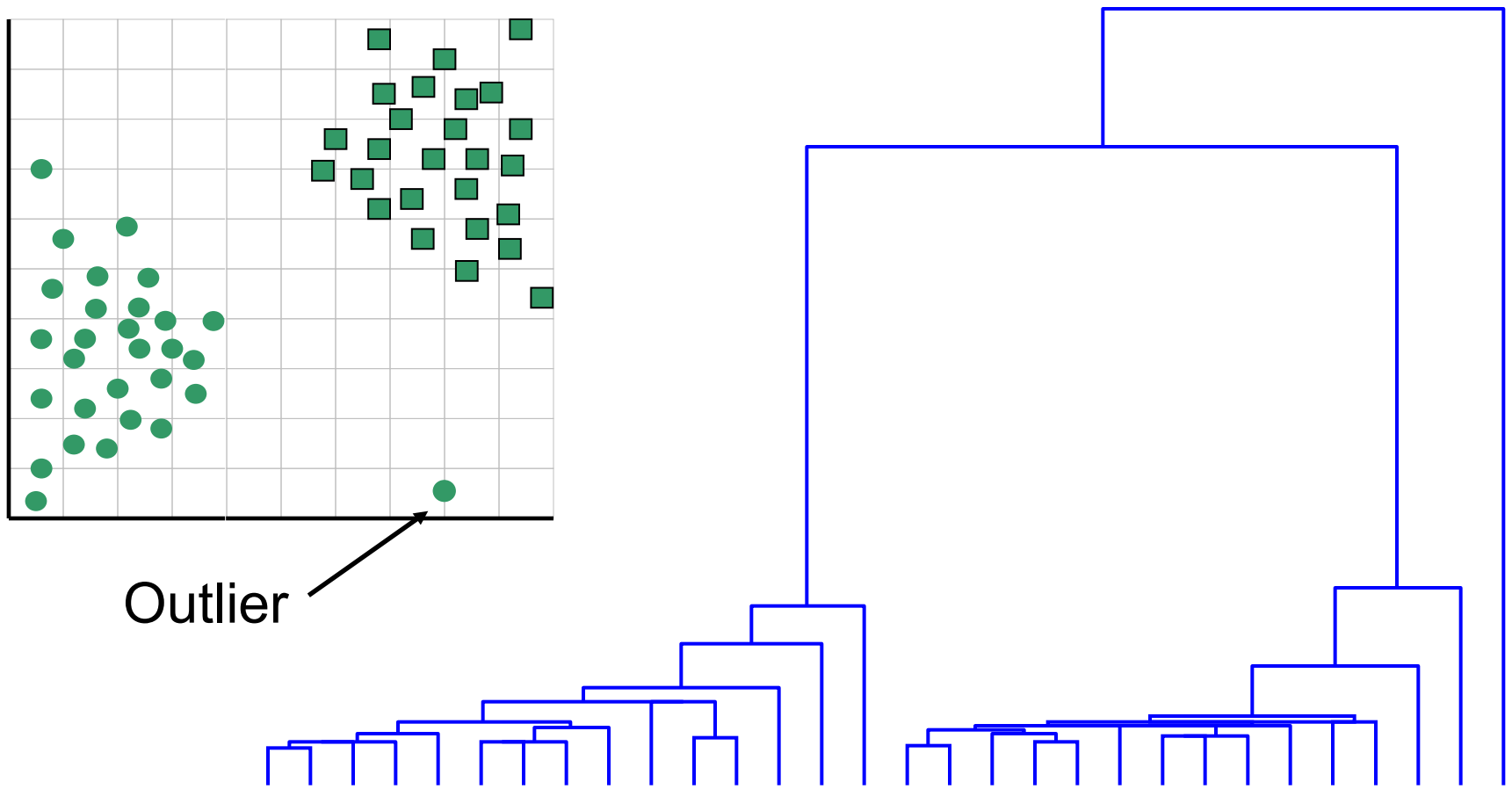
We can look at the dendrogram to determine the “correct” number of clusters.



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

One potential use of a dendrogram: detecting outliers

The single isolated branch is suggestive of a data point that is very different to all others

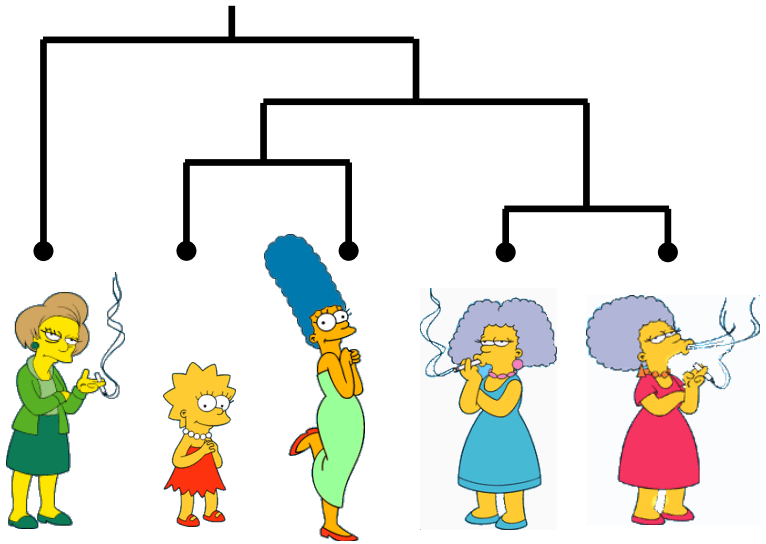


Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

Hierarchical Clustering

The number of dendrograms with n leafs $= (2n - 3)!/[(2^{(n-2)})(n-2)!]$

Number of Leafs	Number of Possible Dendrograms
2	1
3	3
4	15
5	105
...	…
10	34,459,425

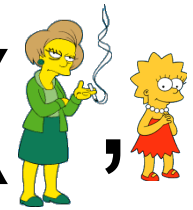


Since we cannot test all possible trees, we will have to heuristic search of all possible trees. We could do this..

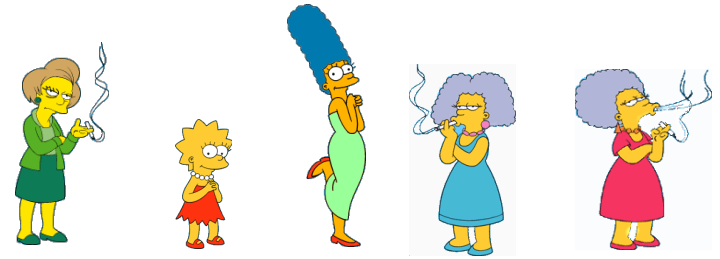
Bottom-Up (agglomerative): Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.

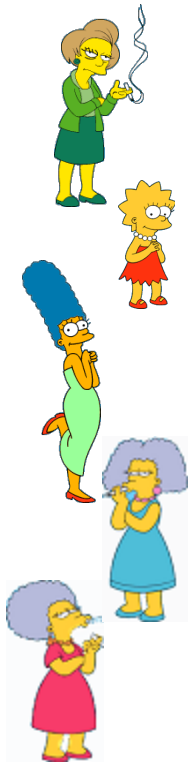
Top-Down (divisive): Starting with all the data in a single cluster, consider every possible way to divide the cluster into two. Choose the best division and recursively operate on both sides.

We begin with a distance matrix which contains the distances between every pair of objects in our database.

D() = 8

D() = 1



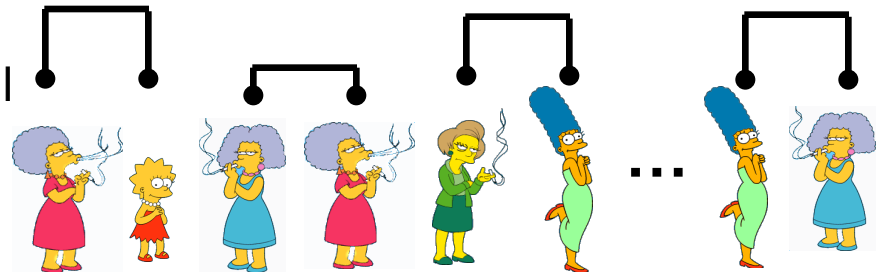


0	8	8	7	7
	0	2	4	4
		0	3	3
			0	1
				0

Bottom-Up (agglomerative): Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.

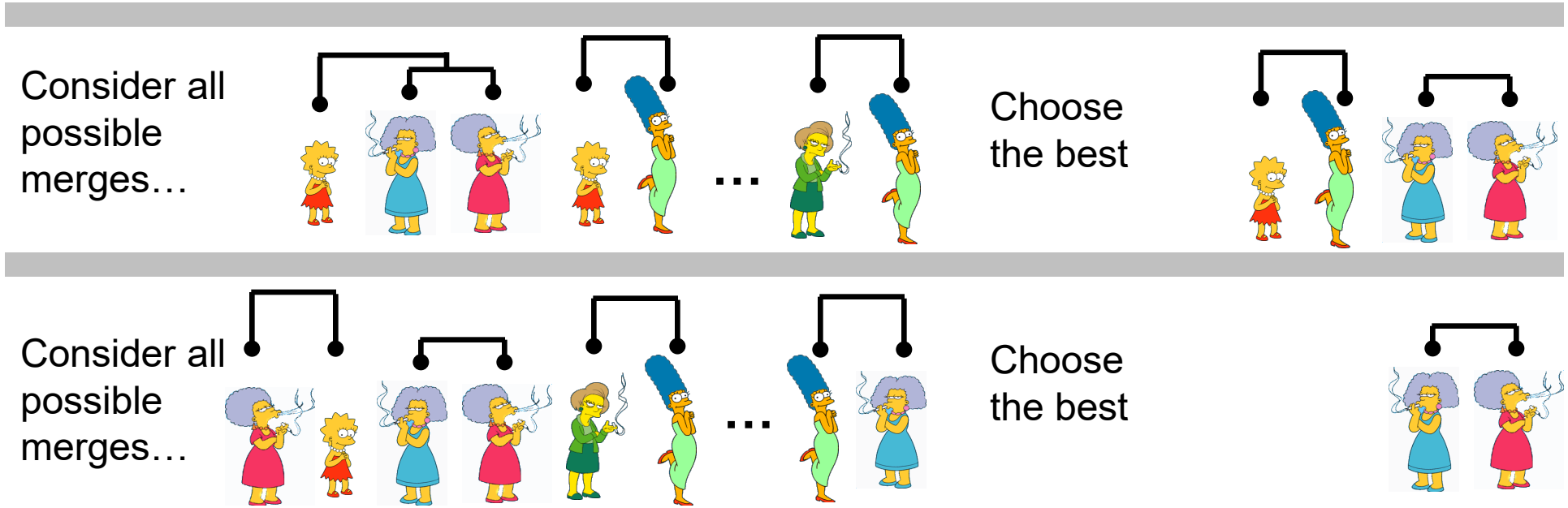
This slide and next 4 based on slides by Eamonn Keogh

Consider all possible merges…

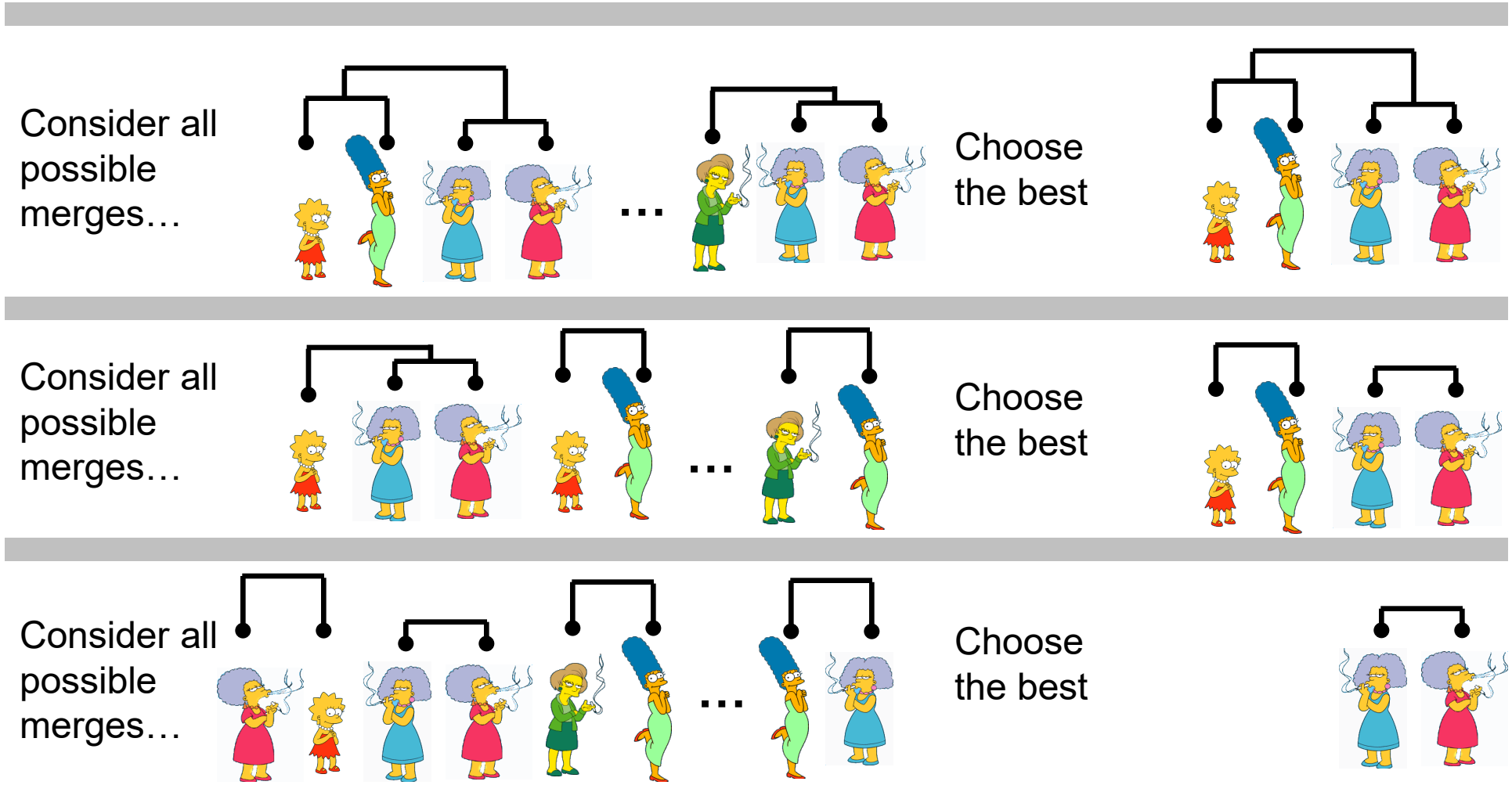


Choose the best

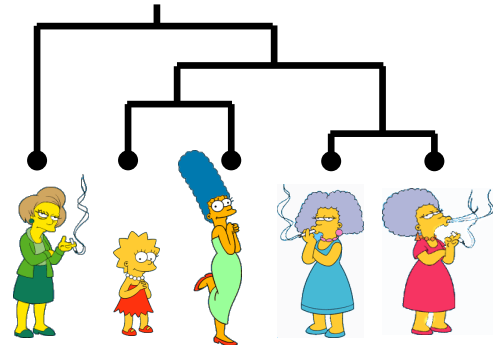
Bottom-Up (agglomerative): Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.



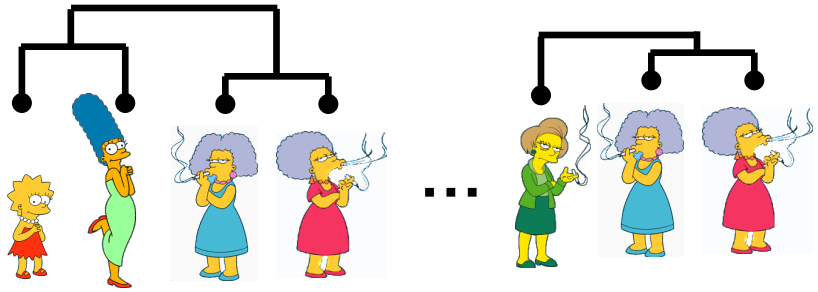
Bottom-Up (agglomerative): Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.



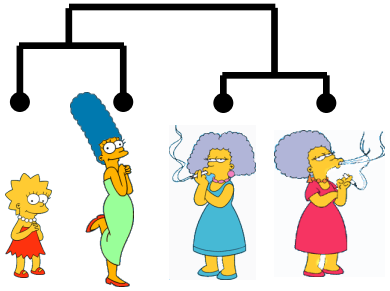
Bottom-Up (agglomerative): Starting with each item in its own cluster, find the best pair to merge into a new cluster. Repeat until all clusters are fused together.



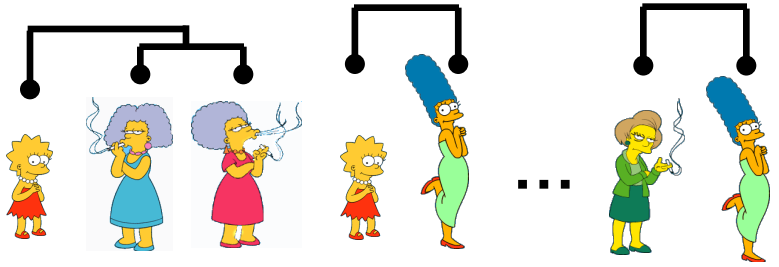
Consider all possible merges…



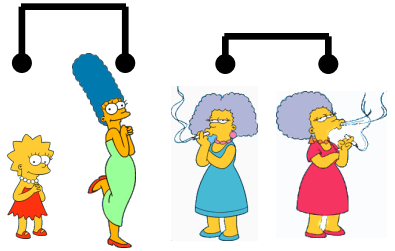
Choose the best



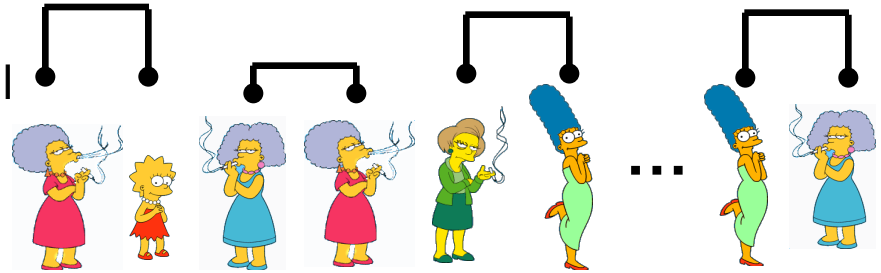
Consider all possible merges…



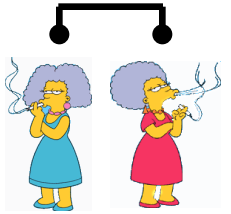
Choose the best



Consider all possible merges…

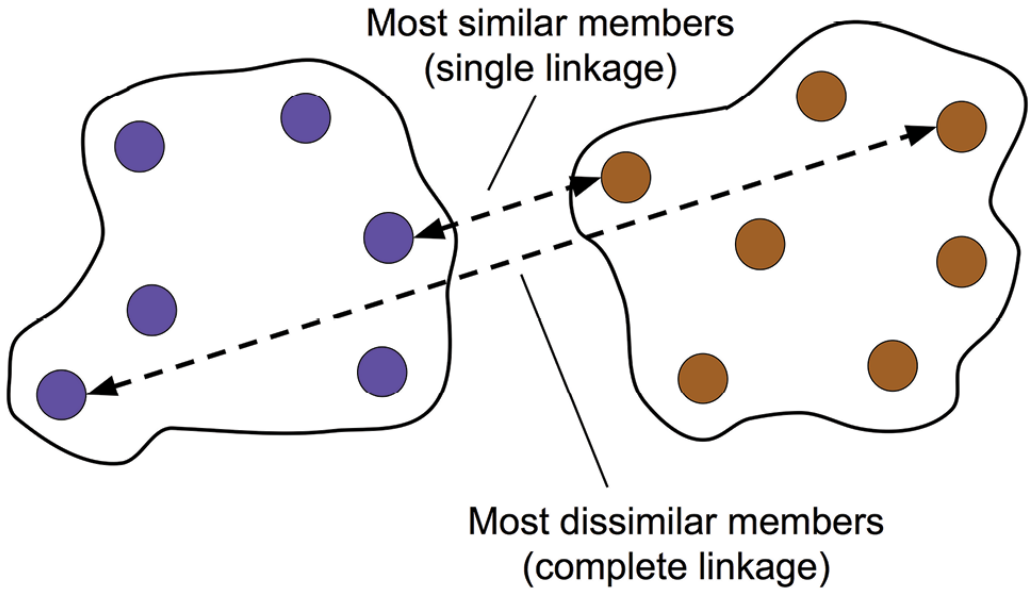


Choose the best



Distances

Figure by S. Raschka



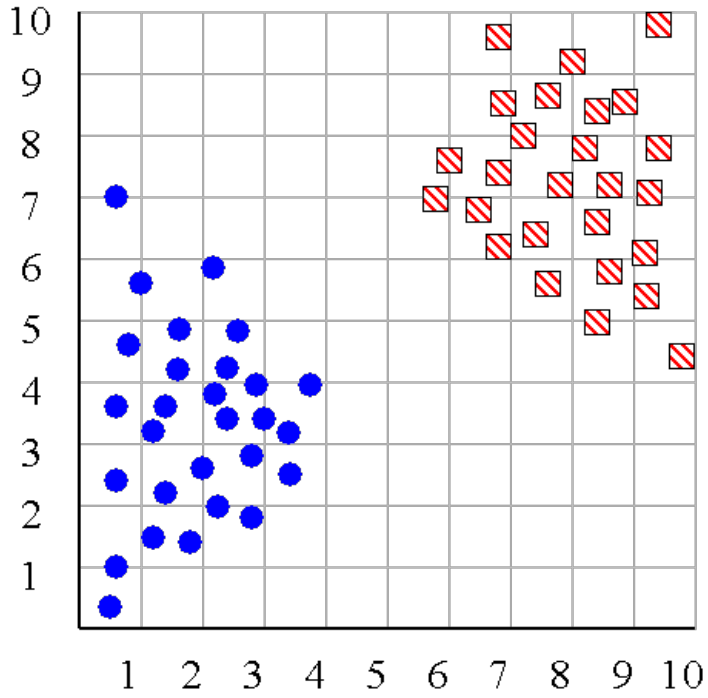
- Have a distance measure on pairs of objects, $d(x, y)$:
- Single linkage: $\text{dist}(\text{A}, \text{B}) = \min_{\text{x}\in A, x'\in B} \text{d}(\text{x}, \text{x}')$
- Complete linkage: $\text{dist}(\text{A}, \text{B}) = \max_{\text{x}\in A, x'\in B} \text{d}(\text{x}, \text{x}')$
- Average linkage: $\text{dist}(\text{A}, \text{B}) = \text{average}_{\text{x}\in A, x'\in B} \text{d}(\text{x}, \text{x}')$
- Ward's method $\text{dist}(\text{A}, \text{B}) = \frac{|A|\,|B|}{|A|+|B|} \|\text{mean}(A) - \text{mean}(B)\|^2$

Slide credit: Maria-Florina Balcan

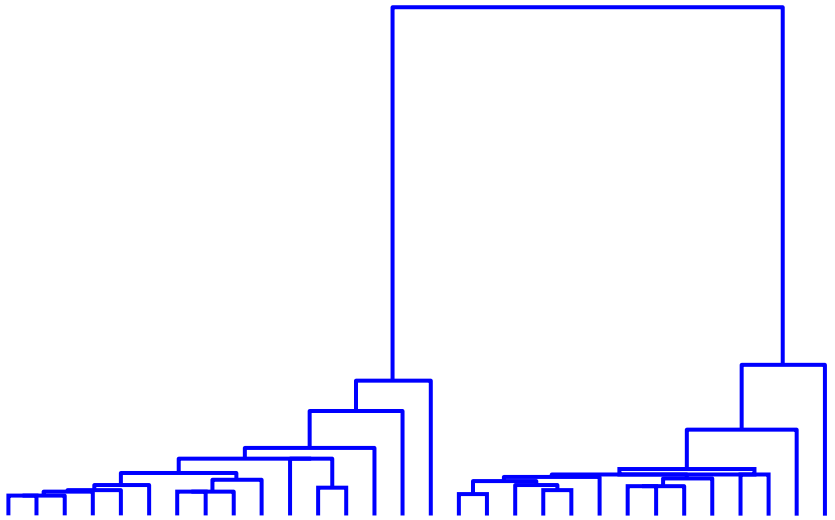
What difference does it make?

- **Single linkage**: $\text{dist}(A, B) = \min_{x \in A, x' \in B} d(x, x')$
  - At any time, distance between any two points in a connected components < r.
- **Complete linkage**: $\text{dist}(A, B) = \max_{x \in A, x' \in B} d(x, x')$
  - Keep max diameter as small as possible at any level
- **Ward's method** $\text{dist}(A, B) = \frac{|A|\,|B|}{|A|+|B|} \|\text{mean}(A) - \text{mean}(B)\|^2$
  - Merge the two clusters such that the increase in k-means cost is as small as possible.
  - Works well in practice

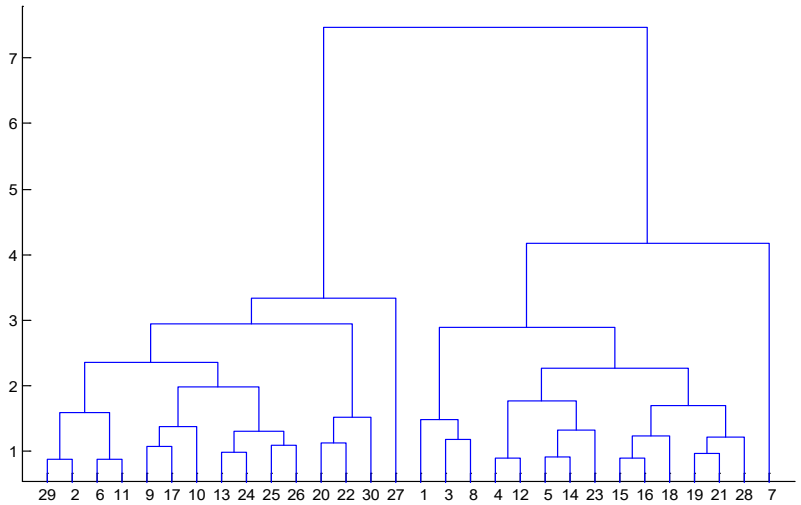
Slide credit: Maria-Florina Balcan



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

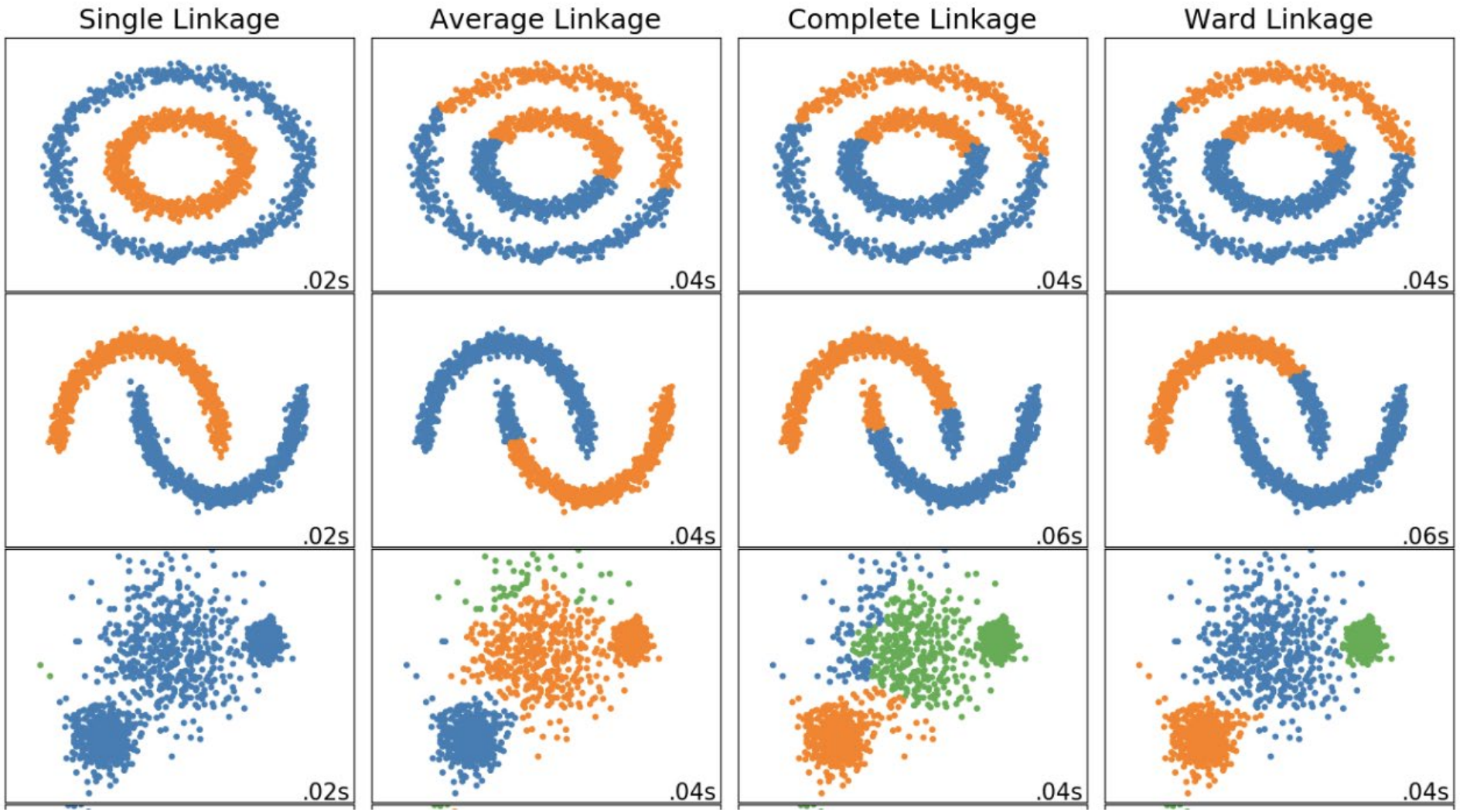


Single linkage



Average linkage

What difference does it make?



Ward linkage measures variance of clusters. The distance between two clusters, A and B, is how much the sum of squares would increase if we merged them.

Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

Hierarchical Clustering Methods Summary

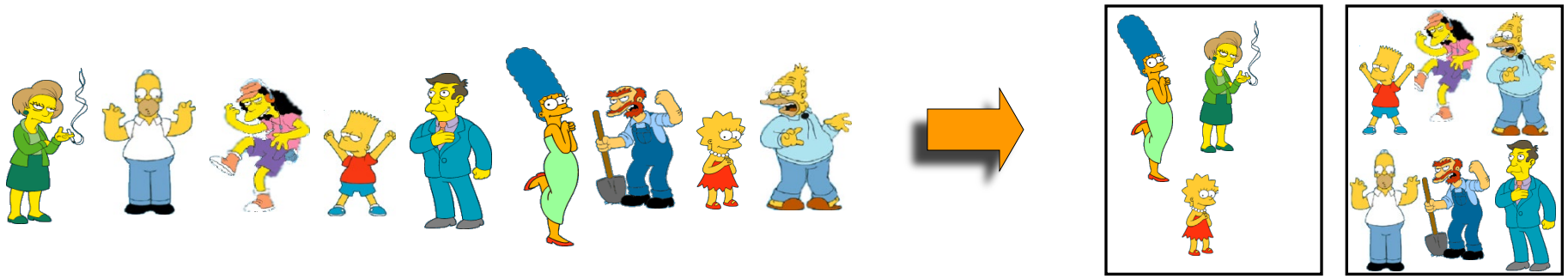
- No need to specify the number of clusters in advance
- Hierarchical nature maps nicely onto human intuition for some domains
- They do not scale well: time complexity of at least $O(n^2)$, where n is the number of total objects
- Like any heuristic search algorithms, local optima are a problem
- Interpretation of results is (very) subjective

Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

Partitional Clustering

- Nonhierarchical, each instance is placed in exactly one of K non-overlapping clusters

- Since only one set of clusters is output, the user normally has to input the desired number of clusters K.

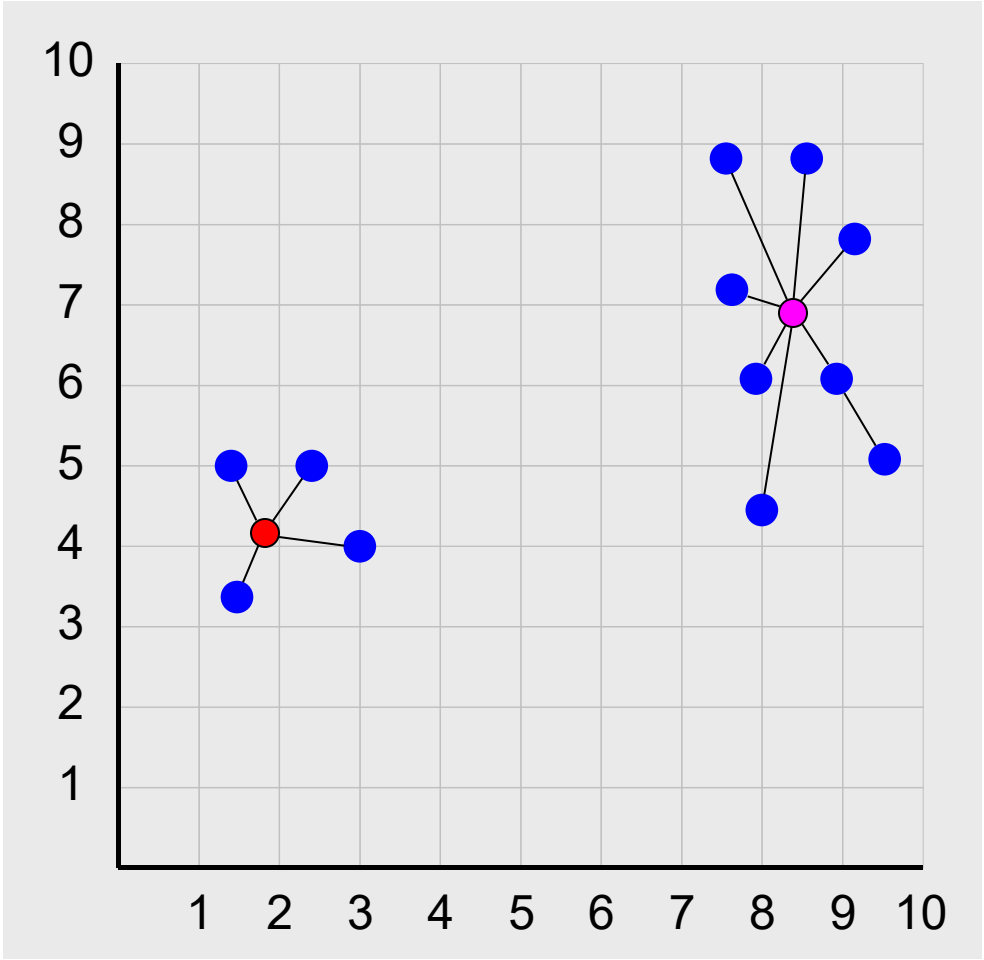


Squared Error

$$se_{K_i} = \sum_{j=1}^{m} \|t_{ij} - C_k\|^2$$

$$se_K = \sum_{j=1}^{k} se_{K_j}$$

Objective Function



Partition Algorithm 1: k-means

1. Decide on a value for *k*.

2. Initialize the *k* cluster centers (randomly, if necessary).

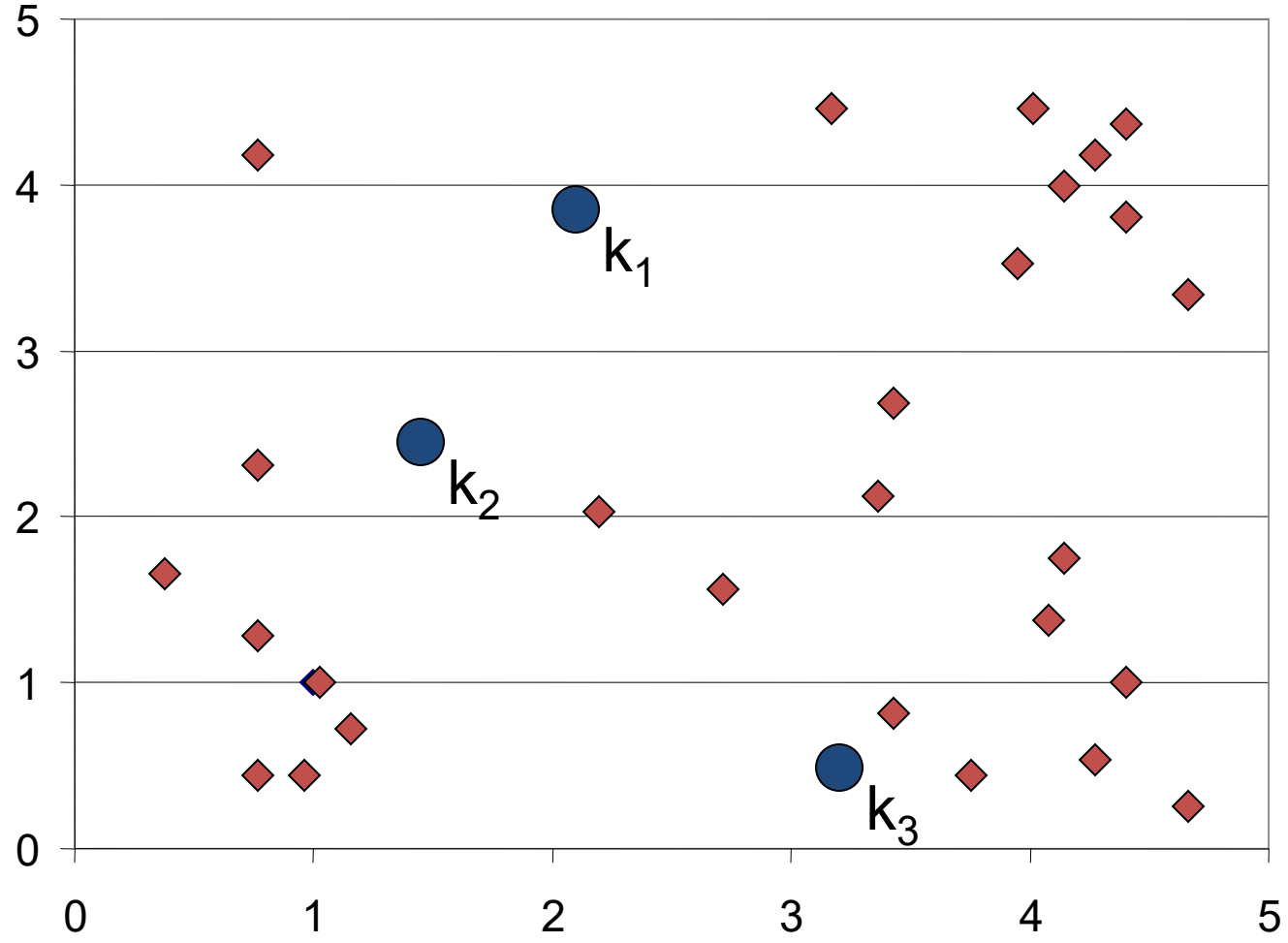
3. Decide the class memberships of the *N* objects by assigning them to the nearest cluster center.

4. Re-estimate the *k* cluster centers, by assuming the memberships found above are correct.

5. If none of the *N* objects changed membership in the last iteration, exit. Otherwise goto 3.

K-means Clustering: Step 1

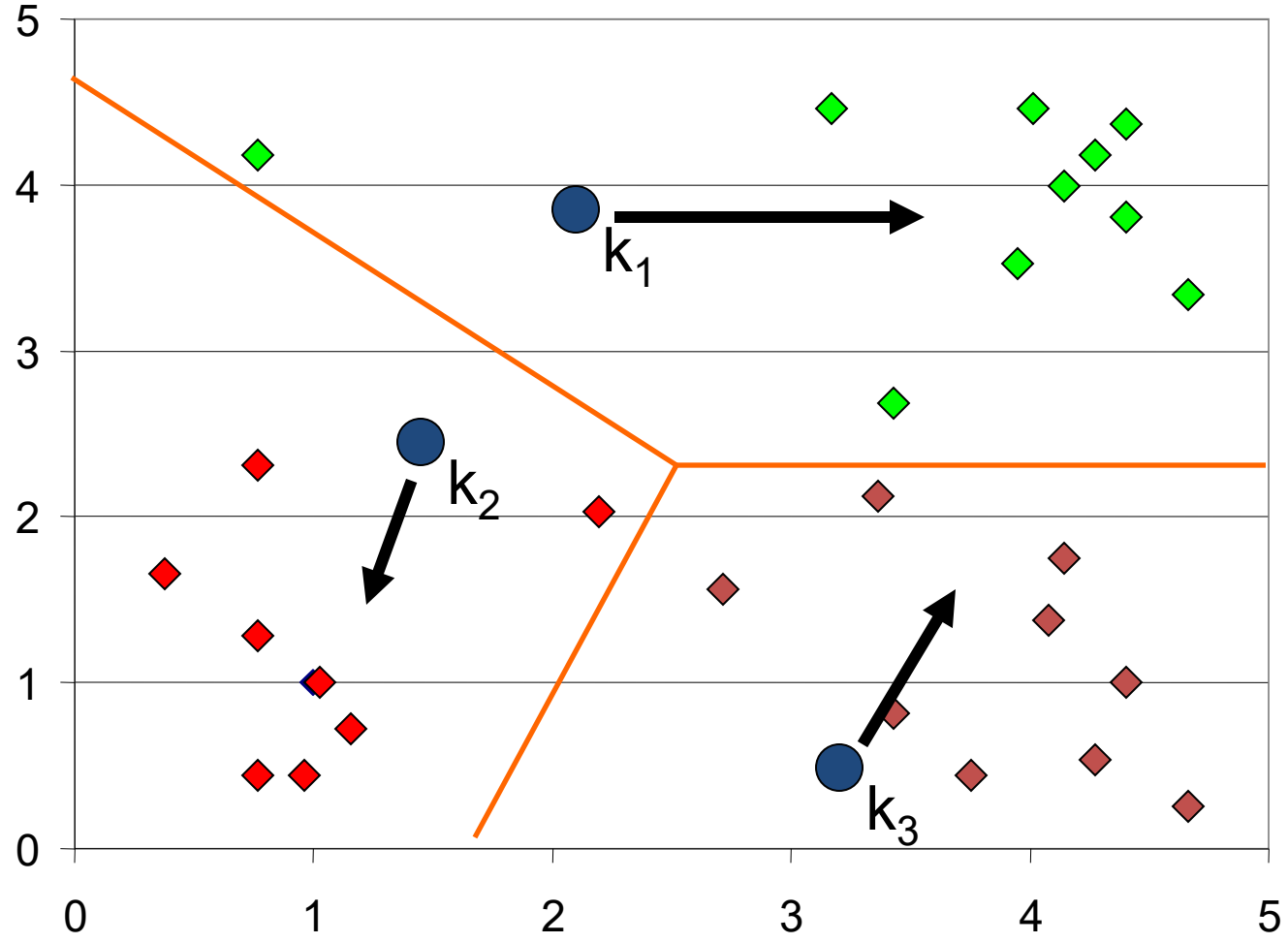
Algorithm: k-means, Distance Metric: Euclidean Distance



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

K-means Clustering: Step 2

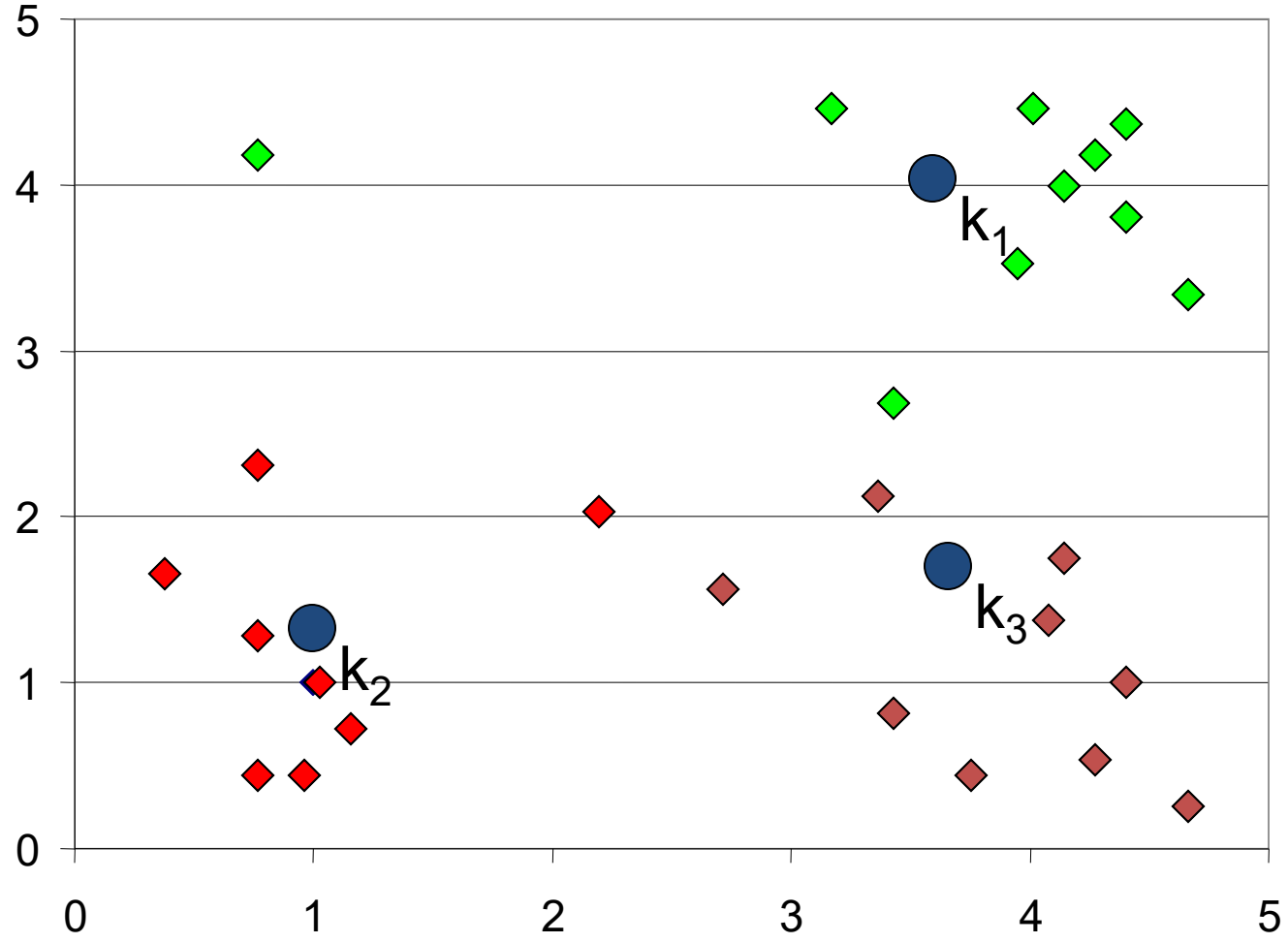
Algorithm: k-means, Distance Metric: Euclidean Distance



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

K-means Clustering: Step 3

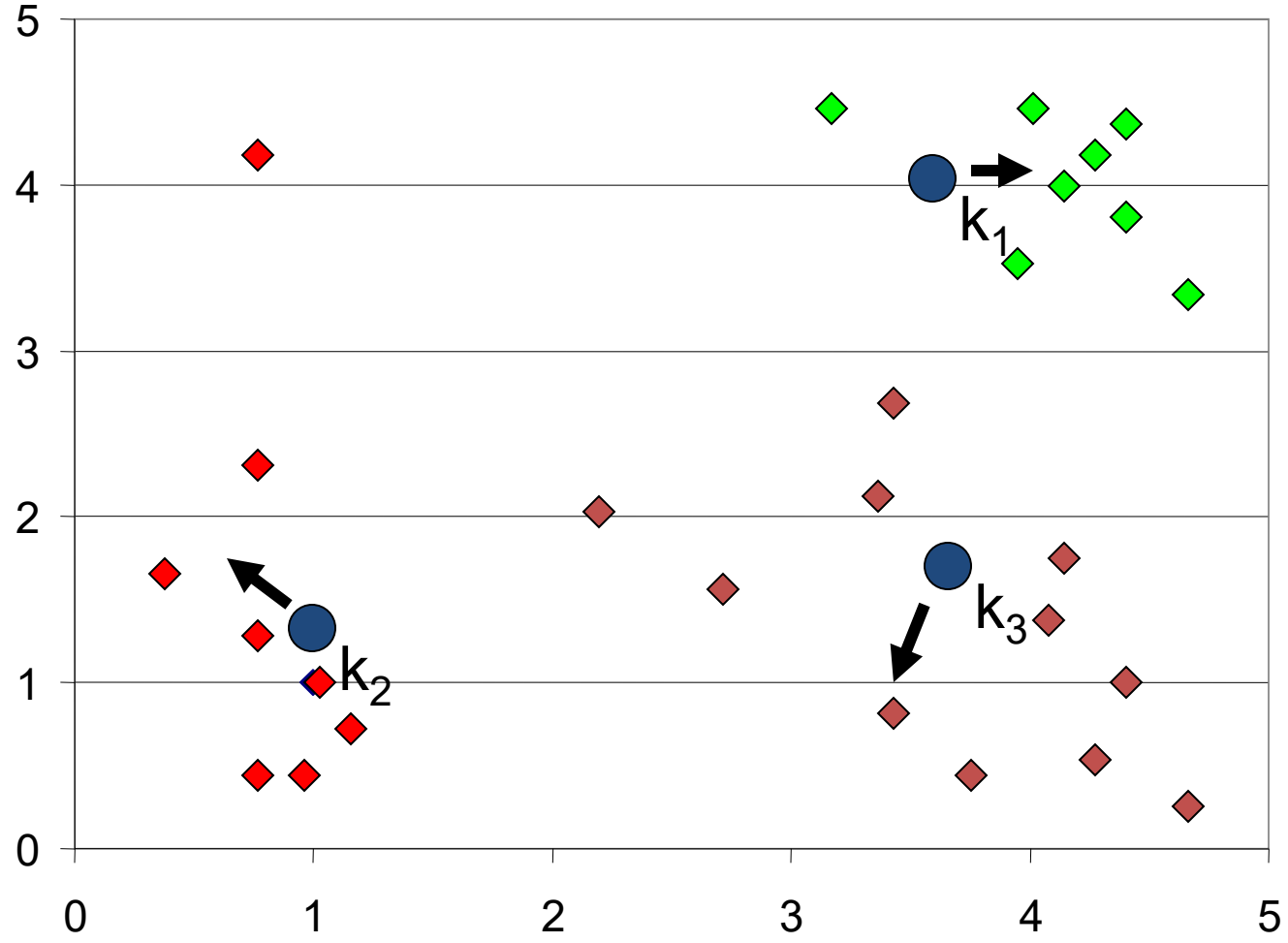
Algorithm: k-means, Distance Metric: Euclidean Distance



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

K-means Clustering: Step 4

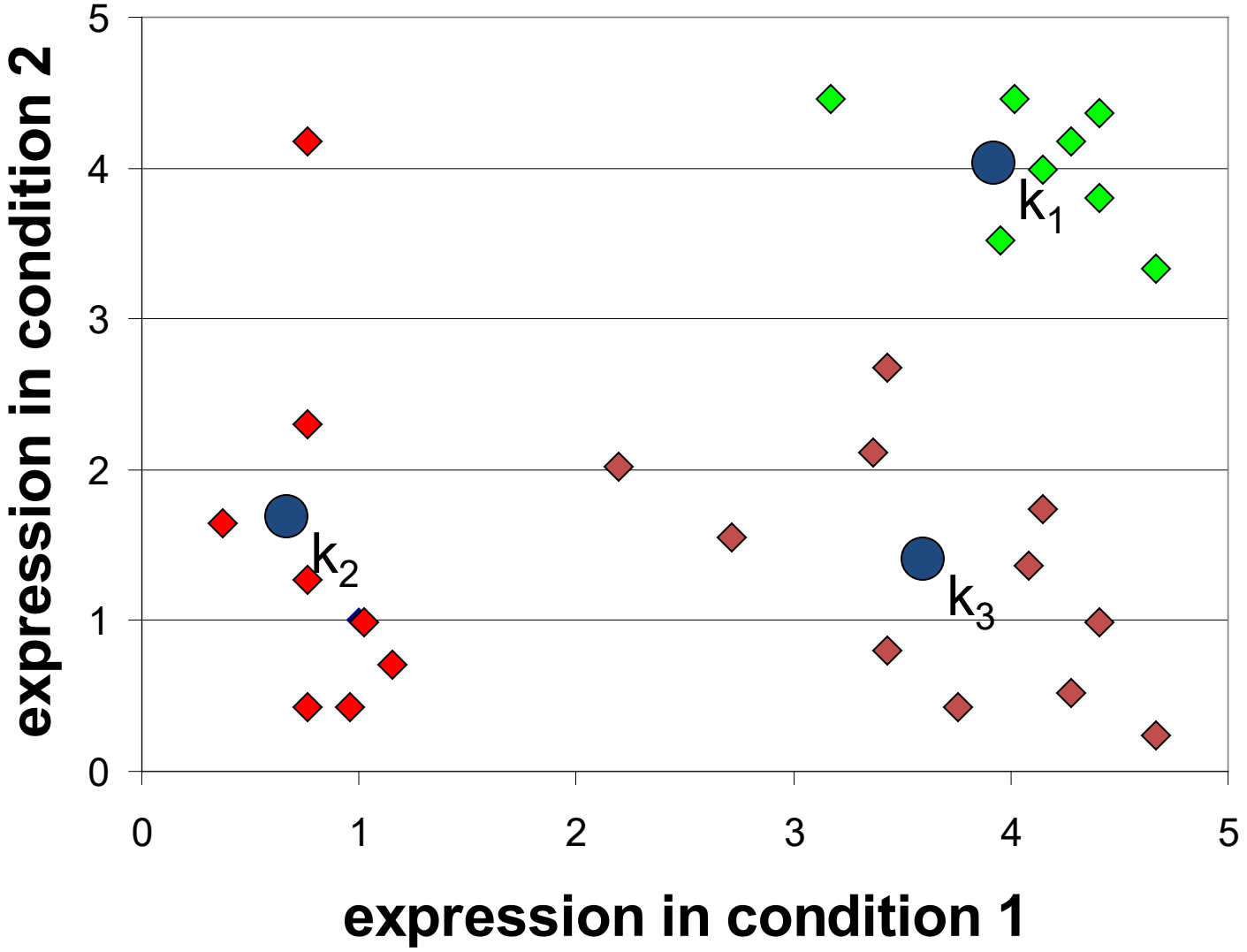
Algorithm: k-means, Distance Metric: Euclidean Distance



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

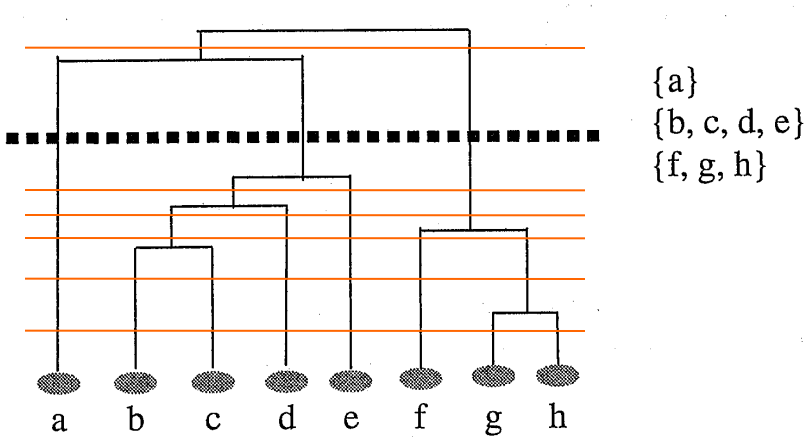
K-means Clustering: Step 5

Algorithm: k-means, Distance Metric: Euclidean Distance



Slide from Eamonn Keogh, from lecture by Carla Brodley, Tufts University

How many clusters to choose?



- Depends on goals
  - May know beforehand how many clusters you want - or at least a range (e.g. 2-10)
  - Could analyze the dendrogram and data after the full clustering to decide which subclustering level is most appropriate for the task at hand
  - Could use automated *cluster validity* metrics to help
- Could do stopping criteria during clustering

How many clusters to choose?

Compactness: members of a cluster are all similar and close together

- One measure of compactness of a cluster is the square distance of the cluster instances compared to the cluster centroid

$$Comp(C) = \sum_{i=1}^{|X_c|} (\mathbf{c} - \mathbf{x}_i)^2$$

- where **c** is the centroid of a cluster C, made up of instances X_c. Lower is better.
- The overall compactness of a particular clustering is just the sum of the compactness of the individual clusters
- Gives a numeric way to compare different clusterings by seeking clusterings which minimize the compactness metric

How many clusters to choose?

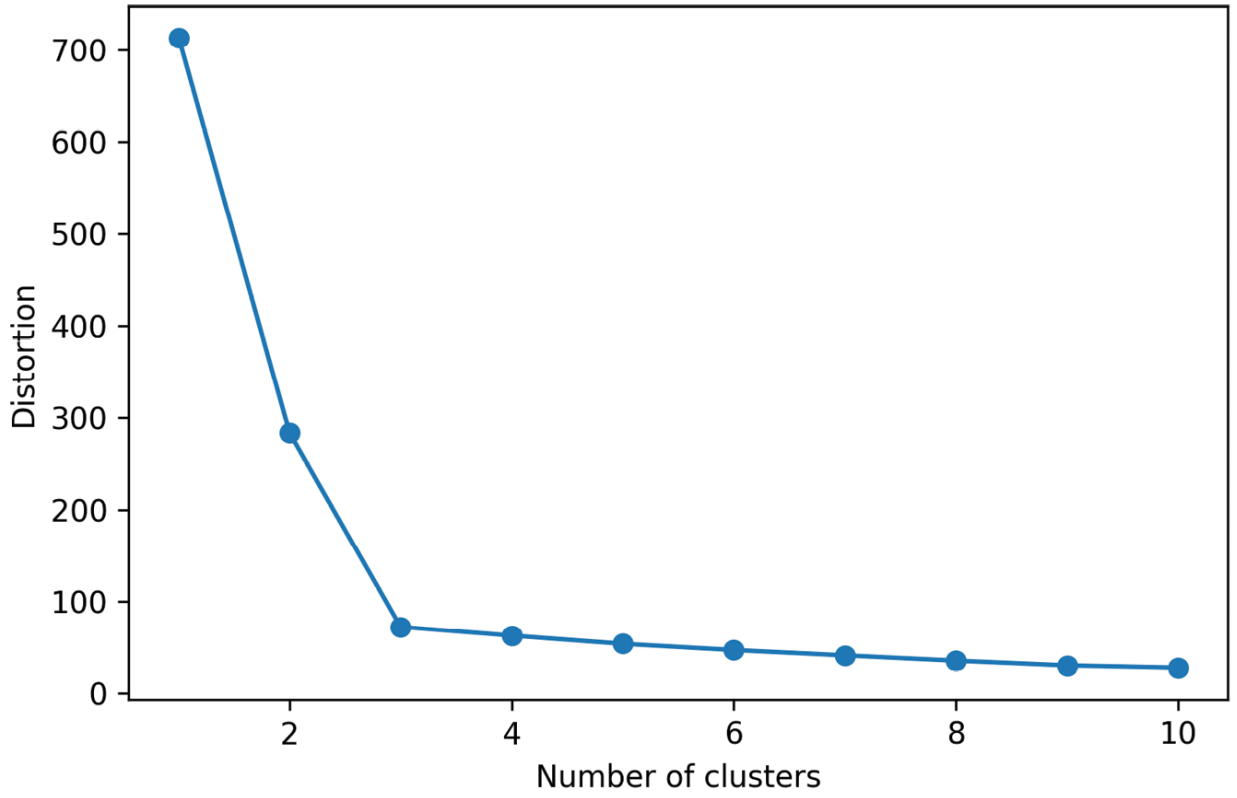
Separability: members of one cluster are sufficiently different from members of another cluster (cluster dissimilarity)

- One measure of the separability of two clusters is their squared distance. The bigger the distance the better.
- $dist_{ij} = (\mathbf{c}_i - \mathbf{c}_j)^2$ where $\mathbf{c}_i$ and $\mathbf{c}_j$ are two cluster centroids
- For a clustering which cluster distances should we compare?
- For each cluster we add in the distance to its closest neighbor cluster

$$Separability = \sum_{i=1}^{|C|} \min_j dist_{ij}(\mathbf{c}_i, \mathbf{c}_j)$$

- We would like to find clusterings where separability is maximized
- separability is usually maximized when there are are very few clusters

Elbow methods



Calculate cluster size as a function of number of clusters

Silhouette

We need techniques that find a balance between inter-cluster similarity and intra-cluster dissimilarity

Silhouette:

- Scores any clustering with an arbitrary number of unique clusters. Clustering can come from any clustering algorithm.
- $a(i)$ = average dissimilarity of instance i to all other instances in the cluster to which i is assigned – **Minimize**
  - Dissimilarity could be Euclidian distance, etc.
- $b(i)$ = the smallest average dissimilarity of instance i to all instances in the closest cluster to $b(i)$ – **Maximize**
- $b(i)$ is smallest for the best different cluster that i could be assigned to – the best cluster that you would move i to if needed

Silhouette

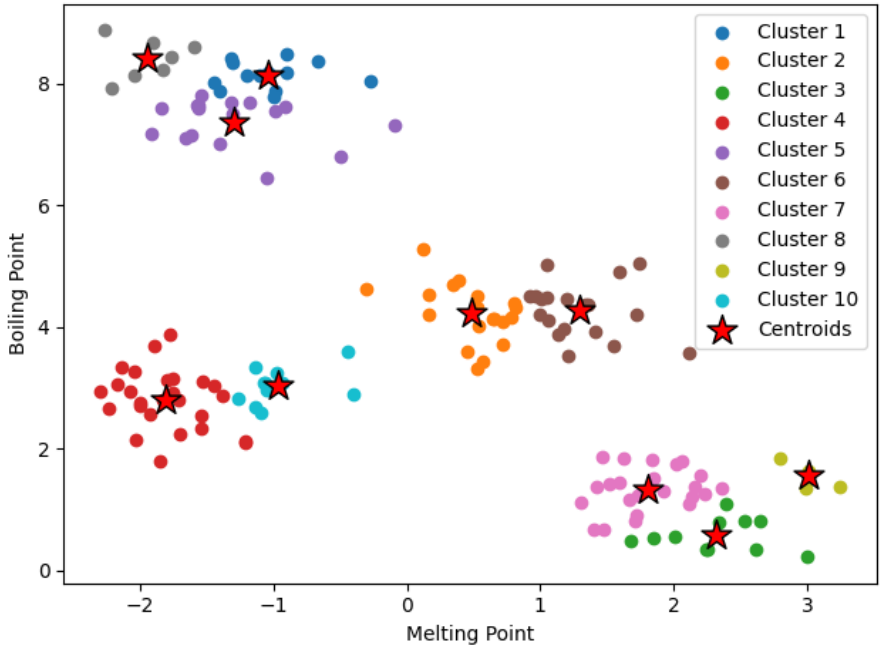
1. Calculate the **cluster cohesion**, $a^{(i)}$, as the average distance between an example, $\boldsymbol{x}^{(i)}$, and all other points in the same cluster.

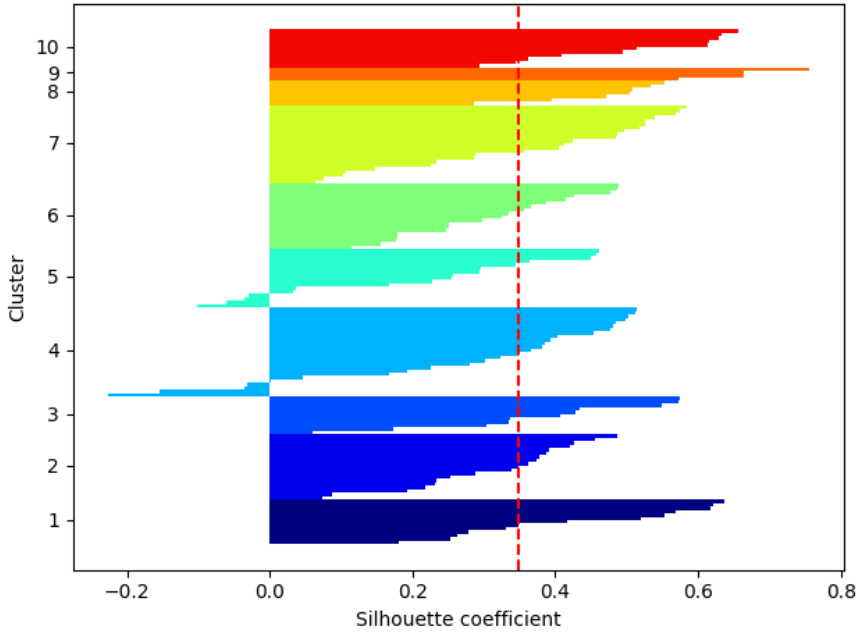
2. Calculate the **cluster separation**, $b^{(i)}$, from the next closest cluster as the average distance between the example, $\boldsymbol{x}^{(i)}$, and all examples in the nearest cluster.

3. Calculate the silhouette, $s^{(i)}$, as the difference between cluster cohesion and separation divided by the greater of the two, as shown here:

$$s^{(i)} = \frac{b^{(i)} - a^{(i)}}{max\{b^{(i)}, a^{(i)}\}}$$

Silhouette





- The quality of a single cluster can be measured by the average silhouette score of its members, (close to 1 is best)
- The quality of a total clustering can be measured by the average silhouette score of all the instances
- To find best clustering, compare total silhouette scores across clusterings with different *k* values and choose the highest

Summary of k-means clustering

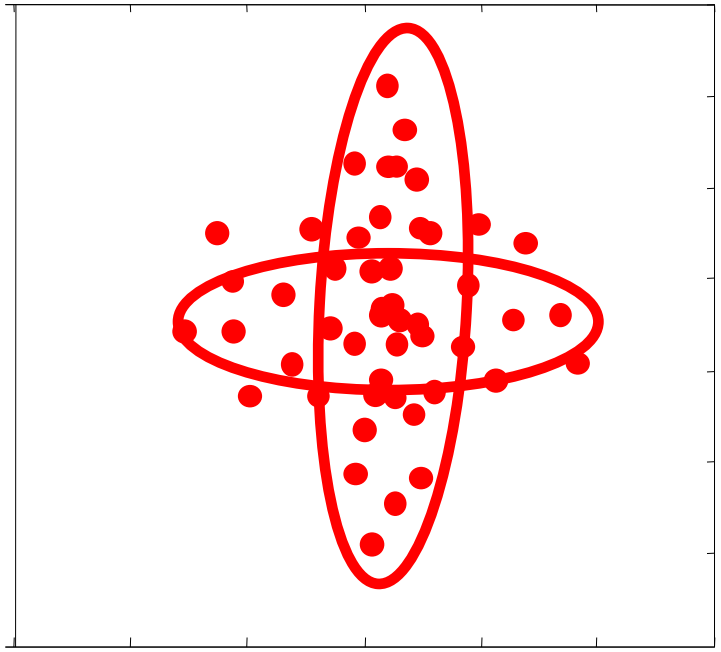
- **Strengths**
  - *Relatively efficient*: $O(tkn)$, where n is number of objects, k is number of clusters, and t is number of iterations. Normally, k, $t << n$.
  - Often terminates at a local optimum

- **Weakness**
  - Applicable only when mean is defined (what about categorical data)?
  - Need to specify k, the number of clusters, in advance
  - Unable to handle noisy data and outliers
  - Not suitable to discover clusters with non-convex shapes
  - Scales matter

Mixture of Gaussians

K-means algorithm

- Assigned each example to exactly one cluster
- What if clusters are overlapping?
  - Hard to tell which cluster is right
  - Maybe we should try to remain uncertain
- Used Euclidean distance
- What if cluster has a non-circular shape?



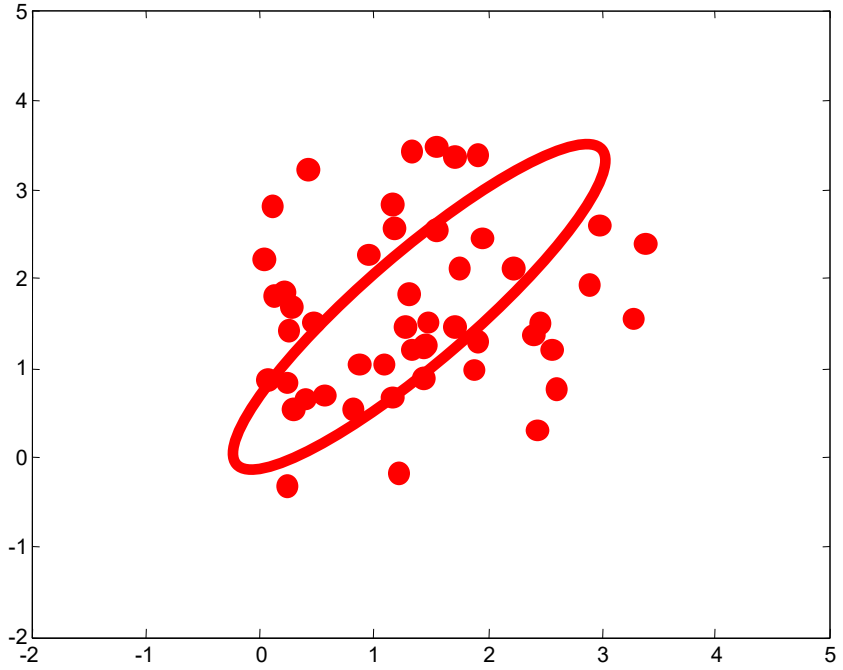
Gaussian mixture models

- Clusters modeled as Gaussian distributions
- EM algorithm: assign data to cluster with some *probability*

(adapted from) Prof. Alexander Ihler

Multivariate Gaussian Model

$$\mathcal{N}(\underline{x}\ ;\ \underline{\mu}, \Sigma) = \frac{1}{(2\pi)^{d/2}} |\Sigma|^{-1/2} \exp\left\{ -\frac{1}{2} (\underline{x} - \underline{\mu})^T \Sigma^{-1} (\underline{x} - \underline{\mu}) \right\}$$



Maximum Likelihood estimates

$$\hat{\mu} = \frac{1}{N} \sum_i x^{(i)}$$

$$\hat{\Sigma} = \frac{1}{N} \sum_i (x^{(i)} - \hat{\mu})^T (x^{(i)} - \hat{\mu})$$

We model each cluster using Gaussian distribution

(adapted from) Prof. Alexander Ihler

Expectation Maximization: E-Step

- Initialize parameters of each cluster: mean μ_c, Covariance Σ_c, size π_c

- **E-step ("Expectation")**
  - For each datum (example) x_i,
  - Compute r_{ic}, the probability that it belongs to cluster c
    - Compute its probability under model c
    - Normalize to sum to one (over clusters c)

$$r_{ic} = \frac{\pi_c \mathcal{N}(x_i \; ; \; \mu_c, \Sigma_c)}{\sum_{c'} \pi_{c'} \mathcal{N}(x_i \; ; \; \mu_{c'}, \Sigma_{c'})}$$

  - If x_i is very likely under the c^{th} Gaussian, it gets high weight
  - Denominator just makes probabilities to sum to one

(adapted from) Prof. Alexander Ihler

Expectation Maximization: M-Step

- Start with assignment probabilities r_{ic}
- Update parameters: mean μ_c, Covariance Σ_c, "size" π_c

- M-step ("Maximization")
  - For each Gaussian cluster x_c,
  - Update its parameters using the (weighted) data points

$$N_c = \sum_i r_{ic}$$

Total responsibility allocated to cluster c

$$\pi_c = \frac{N_c}{N}$$

Fraction of total assigned to cluster c

$$\mu_c = \frac{1}{N_c} \sum_i r_{ic} x_i$$

Weighted mean of assigned data

$$\Sigma_c = \frac{1}{N_c} \sum_i r_{ic} (x_i - \mu_c)^T (x_i - \mu_c)$$

Weighted covariance of assigned data (use new weighted means here)

(adapted from) Prof. Alexander Ihler

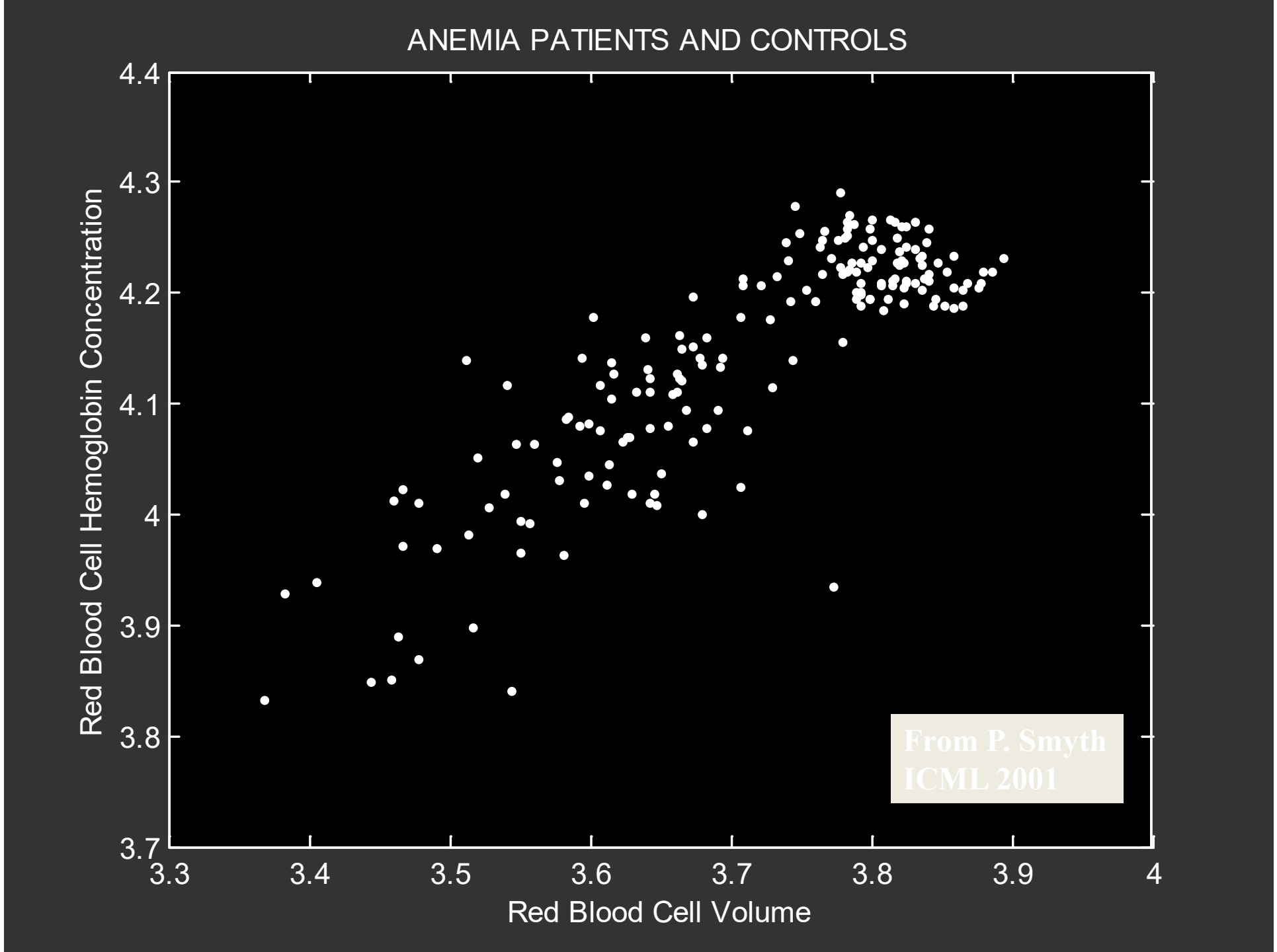
Expectation Maximization

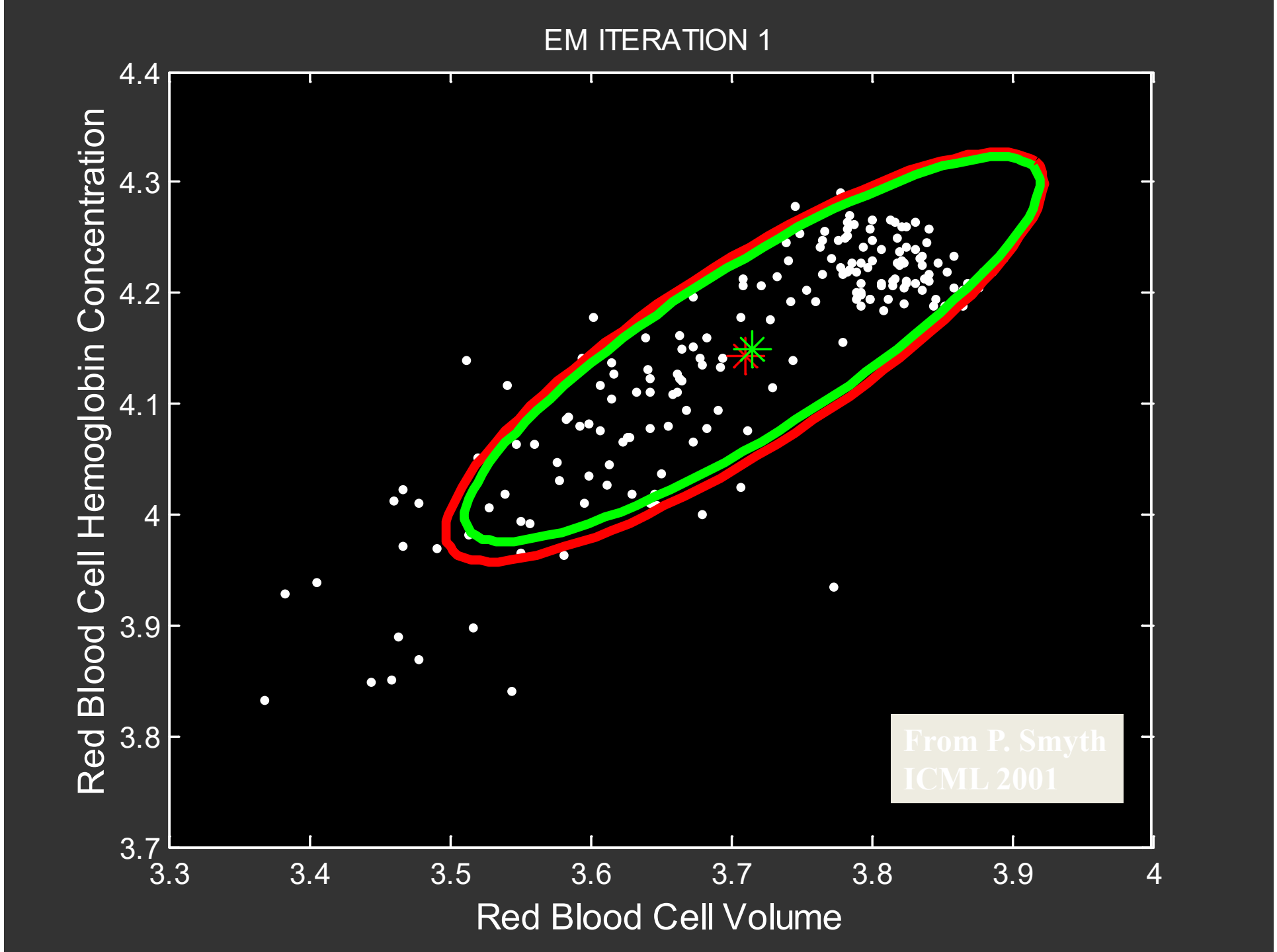
- Each step increases the log-likelihood of our model

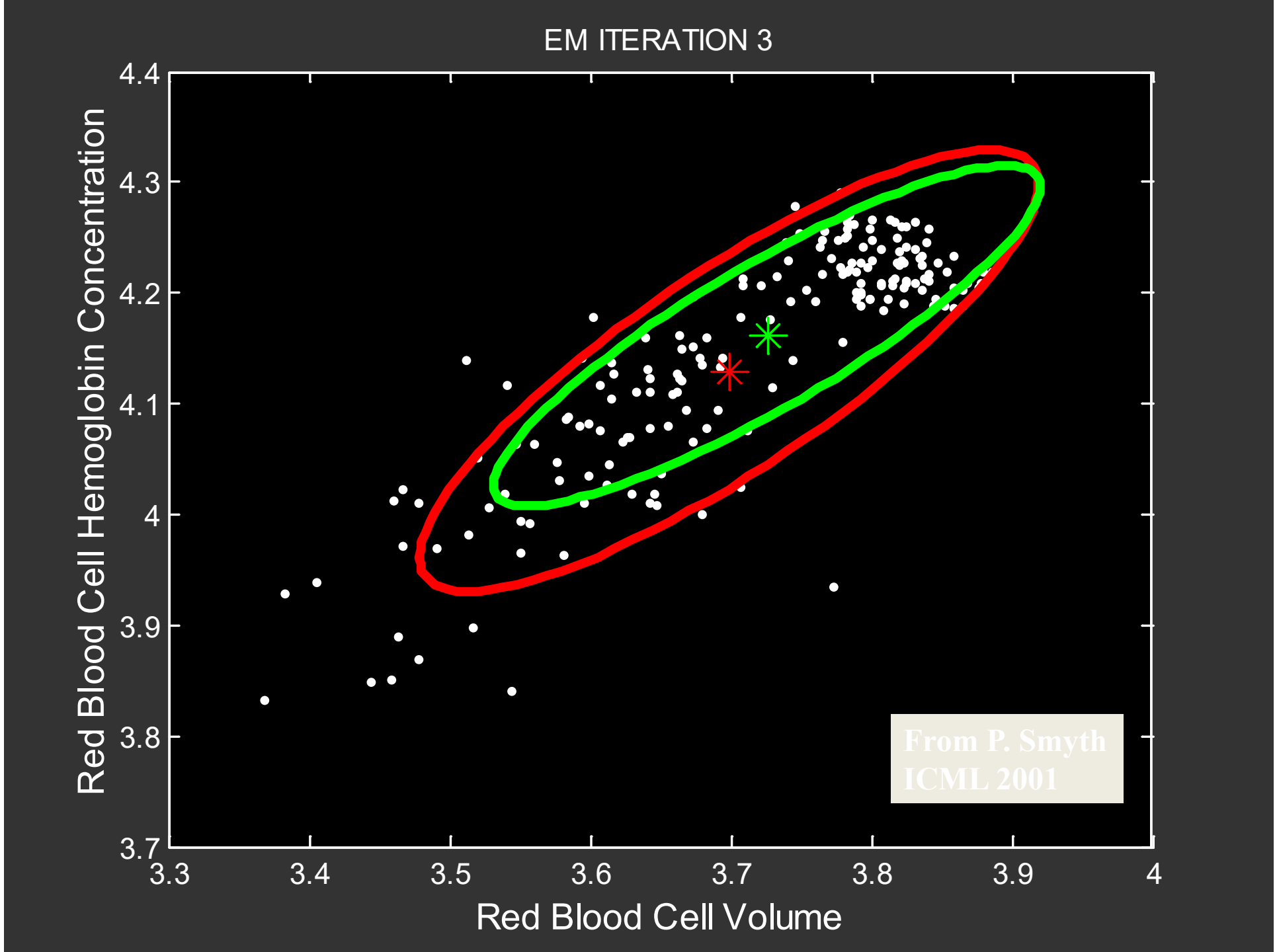
$$\log p(\underline{X}) = \sum_i \log \left[\sum_c \pi_c \, \mathcal{N}(x_i \; ; \; \mu_c, \Sigma_c) \right]$$

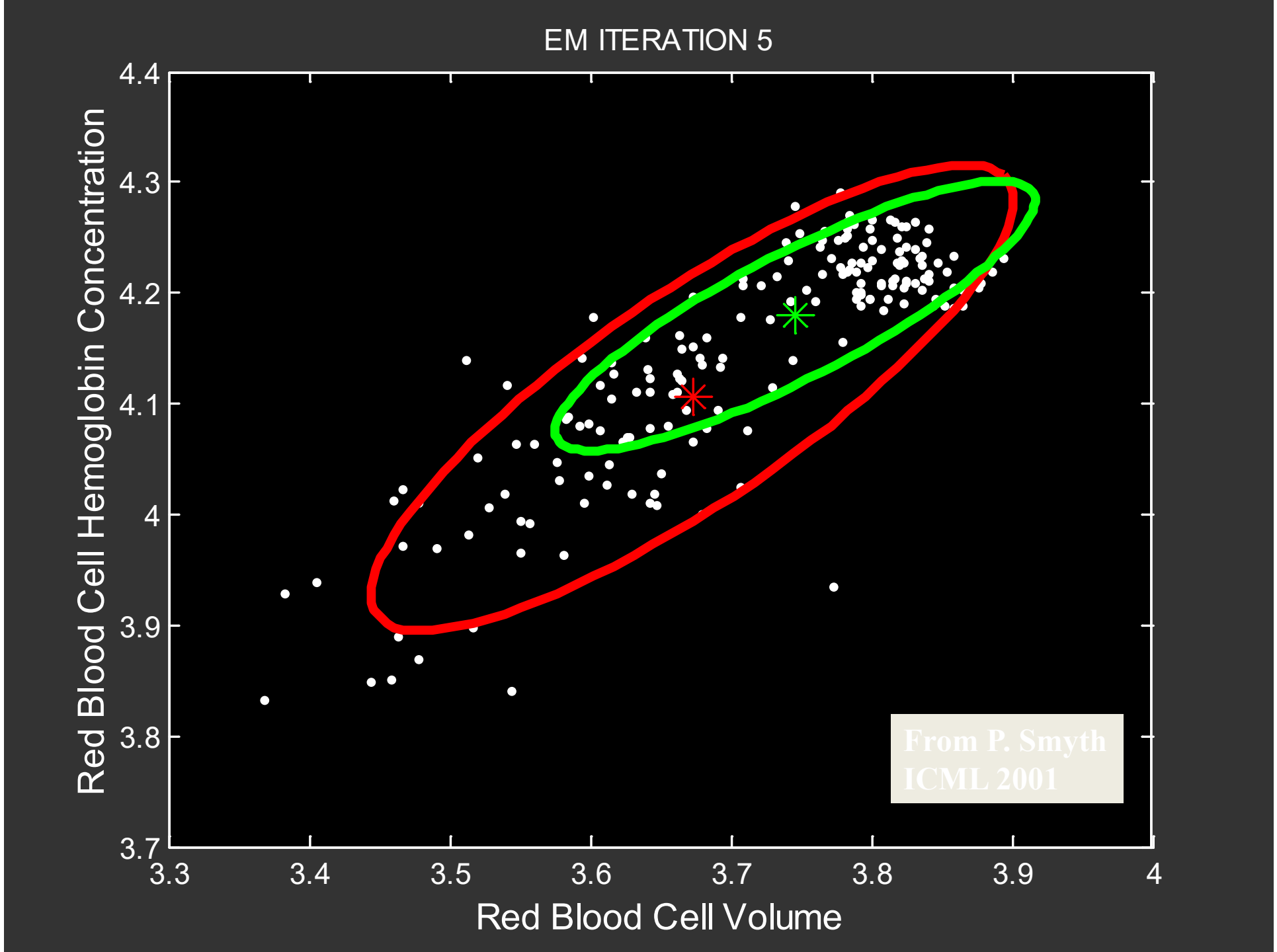
- Iterate until convergence
  - Convergence guaranteed – another ascent method

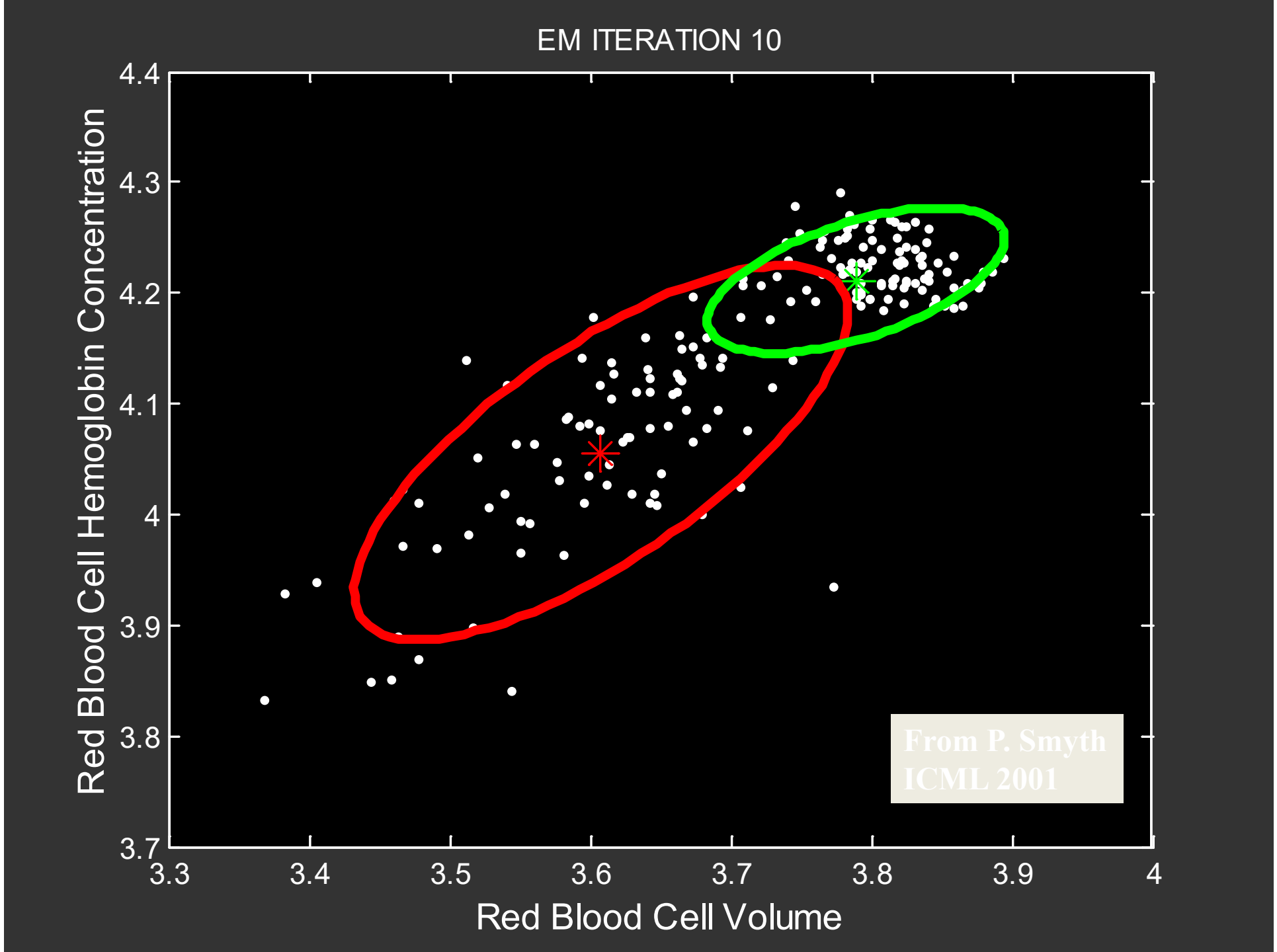
- What should we do
  - If we want to choose a single cluster for an “answer”?
  - With new data we didn’t see during training?

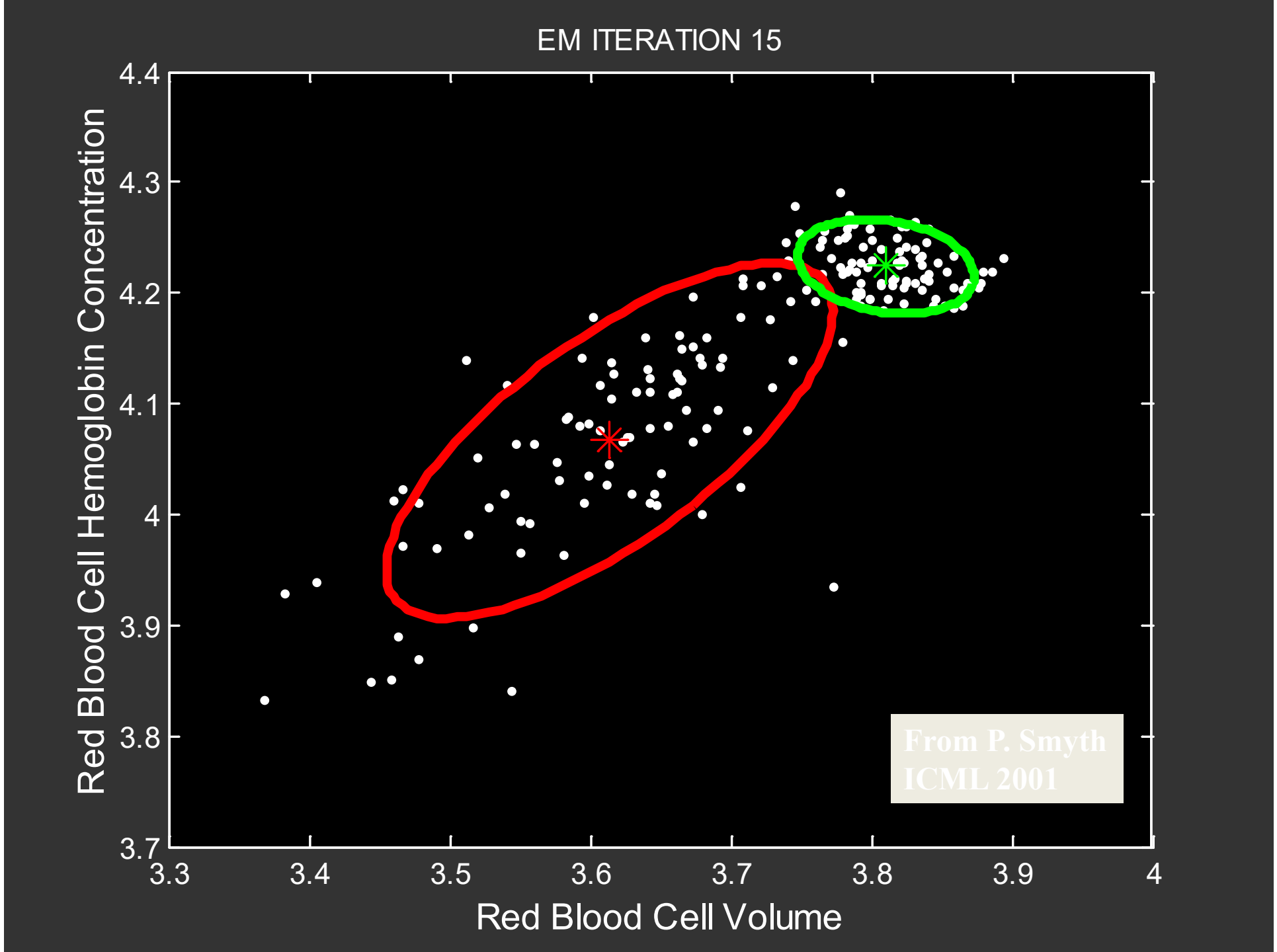

ANEMIA PATIENTS AND CONTROLS
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001

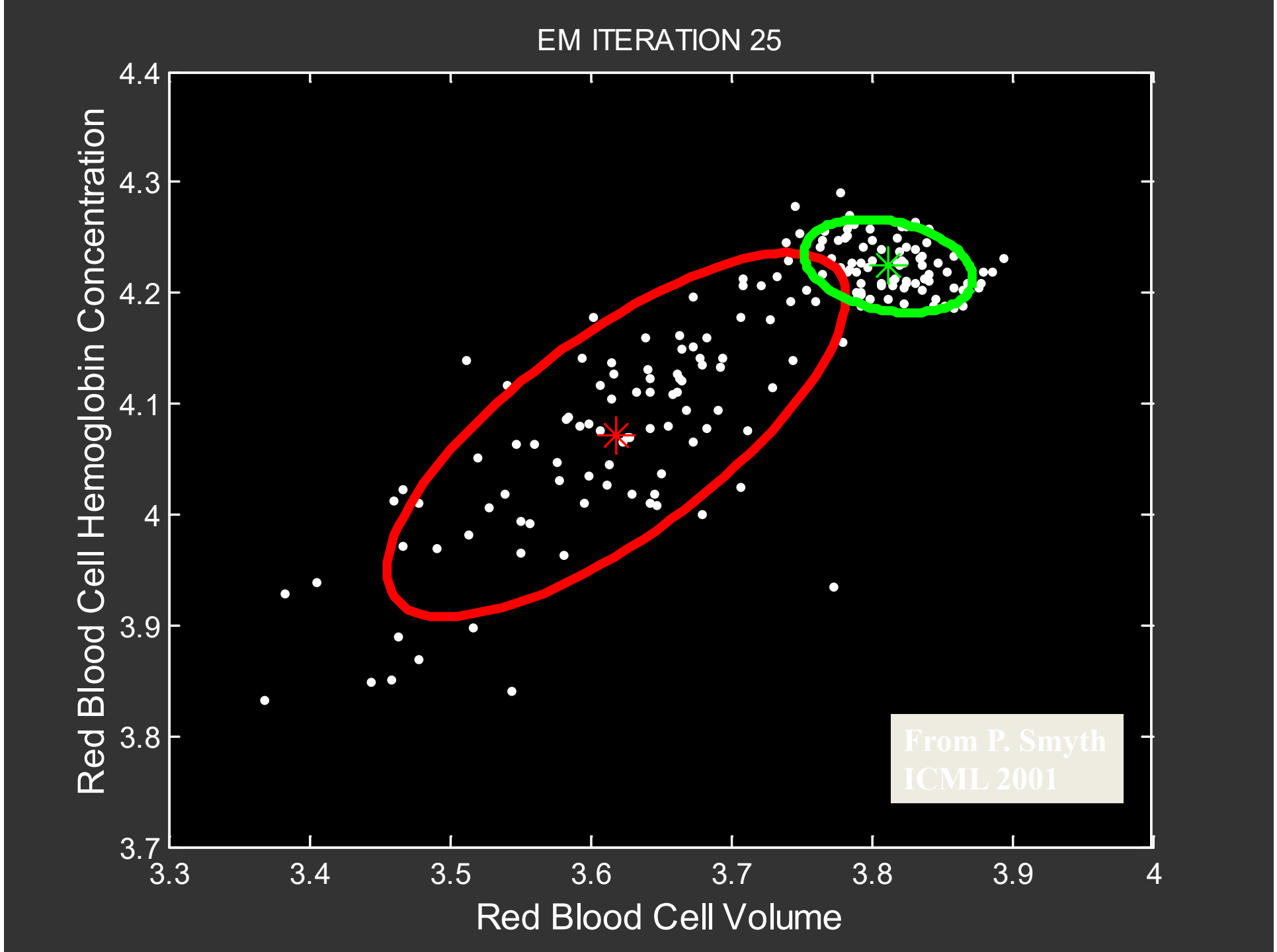

EM ITERATION 1
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001

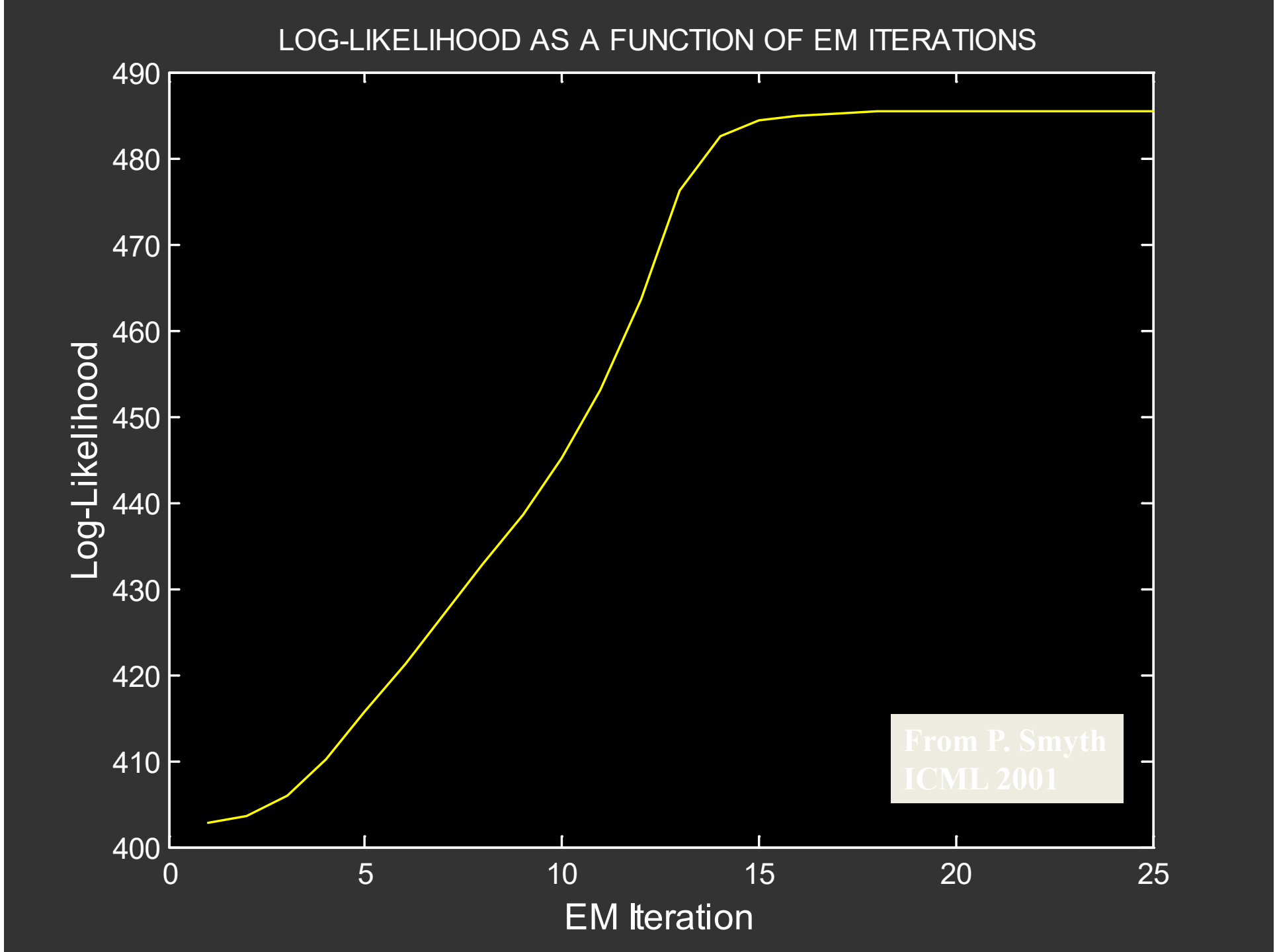

EM ITERATION 3
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


EM ITERATION 5
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


EM ITERATION 10
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


EM ITERATION 15
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


EM ITERATION 25
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


LOG-LIKELIHOOD AS A FUNCTION OF EM ITERATIONS
Log-Likelihood
EM Iteration
490
480
470
460
450
440
430
420
410
400
0
5
10
15
20
25
From P. Smyth
ICML 2001