Expectation Maximization

Brandon Caie and Jonny Coutinho

Intro: Expectation Maximization Algorithm

- EM algorithm provides a general approach to learning in presence of unobserved variables.

- In many practical learning settings, only a subset of relevant features or variables might be observable.
  - Eg: Hidden Markov, Bayesian Belief Networks

Simple Example: Coin Flipping

- Suppose you have 2 coins, A and B, each with a certain bias of landing heads, θ_A , θ_B .

- Given data sets $X_A = \{x_{1,A}, \dots, x_{m_A,A}\}$ and $X_B = \{x_{1,B}, \dots, x_{m_B,B}\}$

  Where $x_{i,j} = \begin{cases} 1\,;\, if\ heads \\ 0\,;\, otherwise \end{cases}$

- No hidden variables – easy solution. $\theta_j = \frac{1}{m_j} \sum_{i=1}^{m_j} x_{i,j}$; sample mean

Simplified MLE

	Coin A	Coin B
H T T T H H T H T H		5 H, 5 T
H H H H T H H H H H	9 H, 1 T	
H T H H H H H T H H	8 H, 2 T	
H T H T T T H H T T		4 H, 6 T
T H H H T H H H T H	7 H, 3 T	
	24 H, 6 T	9 H, 11 T

5 sets, 10 tosses per set

$$\hat{\theta}_A = \frac{24}{24+6} = 0.80$$

$$\hat{\theta}_B = \frac{9}{9+11} = 0.45$$

Goal: determine coin parameters without knowing the identity of each data set's coin.

Solution: Expectation-maximization

Coin Flip With hidden variables

- What if you were given the same dataset of coin flip results, but no coin identities defining the datasets?

Here: $X = \{x_1, \dots x_m\}$; the observed variable

$$Z = \begin{Bmatrix} z_{1,1} & \cdots & z_{m,1} \\ \cdots & z_{i,j} & \cdots \\ z_{1,k} & \cdots & z_{m,k} \end{Bmatrix} \quad \text{where } z_{i,j} = \begin{Bmatrix} 1\,;\, if\ x_i\ is\ from\ j^{th}\ coin \\ 0;\, otherwise \end{Bmatrix}$$

But Z is not known. (Ie: ‘hidden’ / ‘latent’ variable)

EM Algorithm

0) Initialize some arbitrary hypothesis of parameter values (θ):

$\theta = \{\theta_1, \ldots, \theta_k\}$ coin flip example: $\theta = \{\theta_A, \theta_B\}$ = {0.6, 0.5}

1) Expectation (E-step)

$$E[z_{i,j}] = \frac{p(x = x_i | \theta = \theta_j)}{\sum_{n=1}^{k} p(x = x_i | \theta = \theta_n)}$$

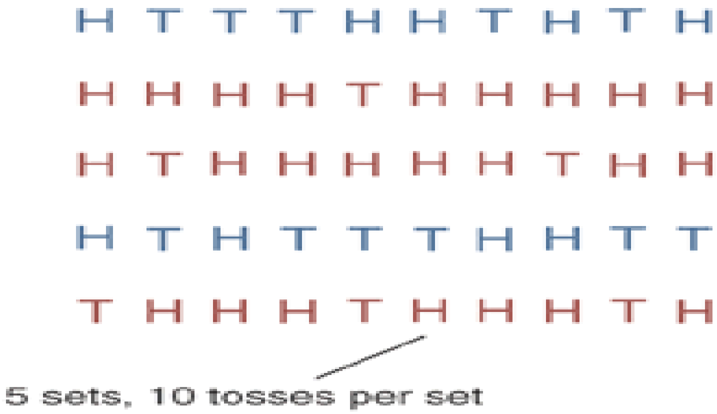
2) Maximization (M-step)

$$\theta_j = \frac{\sum_{i=1}^{m} E[z_{i,j}] x_i}{\sum_{i=1}^{m} E[z_{i,j}]}$$

If $z_{i,j}$ is known:

$$\theta_j = \frac{\sum_{i=1}^{m_j} x_i}{m_j}$$

EM- Coin Flip example



- Initialize θ_A and θ_B to chosen value
  - Ex: θ_A=0.6, θ_B= 0.5

- Compute a probability distribution of possible completions of the data using current parameters

EM- Coin Flip example

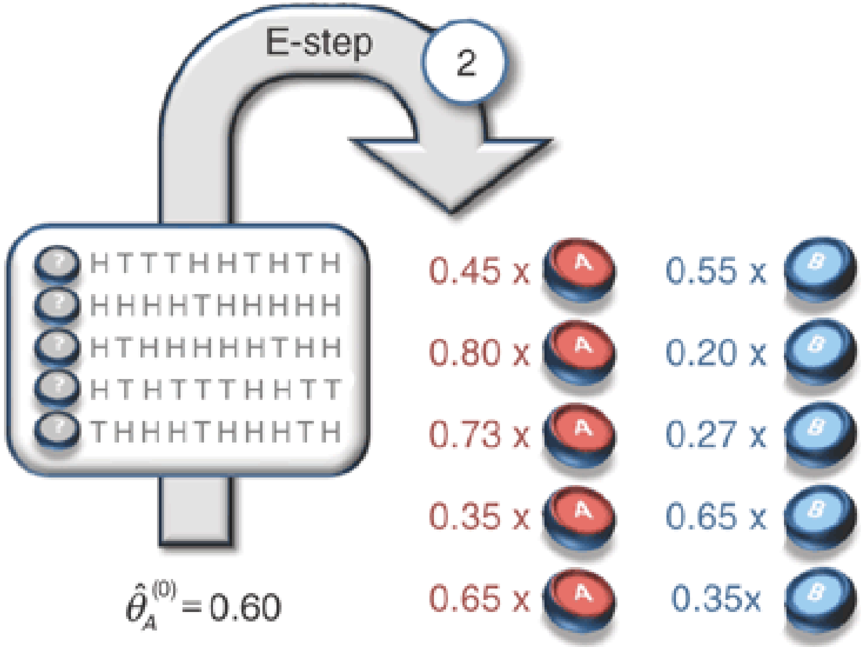
Set 1 H T T T H H T H T H

- What is the probability that I observe 5 heads and 5 tails in coin A and B *given* the initializing parameters θ_A=0.6, θ_B= 0.5?
- Compute likelihood of set 1 coming from coin A or B using the binomial distribution with mean probability θ on *n* trials with *k* successes

$$p(k) = \binom{n}{k}\theta^k(1-\theta)^{n-k}$$

- Likelihood of “A”=0.00079
- Likelihood of “B”=0.00097
- Normalize to get probabilities→ A=0.45, B=0.55

The E-step



$\hat{\theta}_A^{(0)} = 0.60$

$\hat{\theta}_B^{(0)} = 0.50$

Coin A	Coin B
≈ 2.2 H, 2.2 T	≈ 2.8 H, 2.8 T
≈ 7.2 H, 0.8 T	≈ 1.8 H, 0.2 T
≈ 5.9 H, 1.5 T	≈ 2.1 H, 0.5 T
≈ 1.4 H, 2.1 T	≈ 2.6 H, 3.9 T
≈ 4.5 H, 1.9 T	≈ 2.5 H, 1.1 T
≈ 21.3 H, 8.6 T	≈ 11.7 H, 8.4 T

The M-step

Coin A	Coin B
≈ 2.2 H, 2.2 T	≈ 2.8 H, 2.8 T
≈ 7.2 H, 0.8 T	≈ 1.8 H, 0.2 T
≈ 5.9 H, 1.5 T	≈ 2.1 H, 0.5 T
≈ 1.4 H, 2.1 T	≈ 2.6 H, 3.9 T
≈ 4.5 H, 1.9 T	≈ 2.5 H, 1.1 T
≈ 21.3 H, 8.6 T	≈ 11.7 H, 8.4 T

$$\hat{\theta}_A^{(1)} \approx \frac{21.3}{21.3 + 8.6} \approx 0.71$$

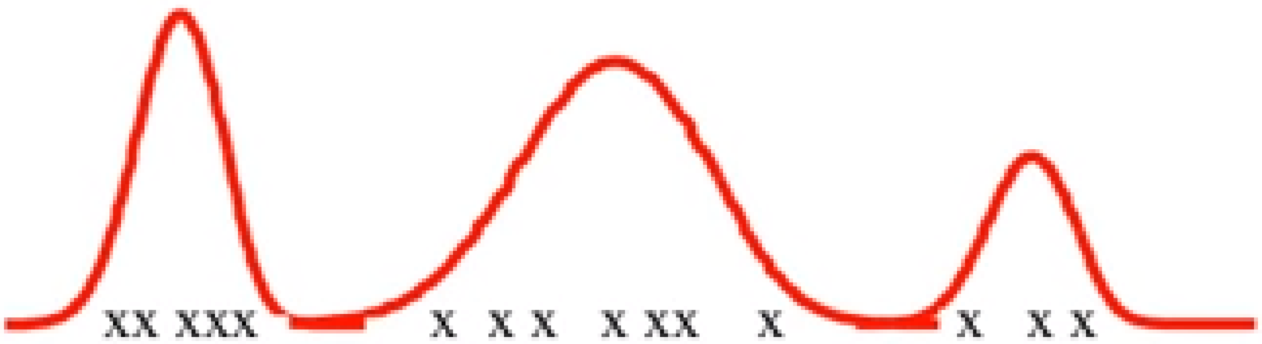
$$\hat{\theta}_B^{(1)} \approx \frac{11.7}{11.7 + 8.4} \approx 0.58$$

Summary

1. Choose starting parameters
2. Estimate probability using these parameters that each data set (x_i) came from j^{th} coin ($E[z_{i,j}]$)
3. Use these probability values ($E[z_{i,j}]$) as weights on each data point when computing a new θ_j to describe each distribution
4. Summate these expected values, use maximum likelihood estimation to derive new parameter values to repeat process

Gaussian Mixture Models

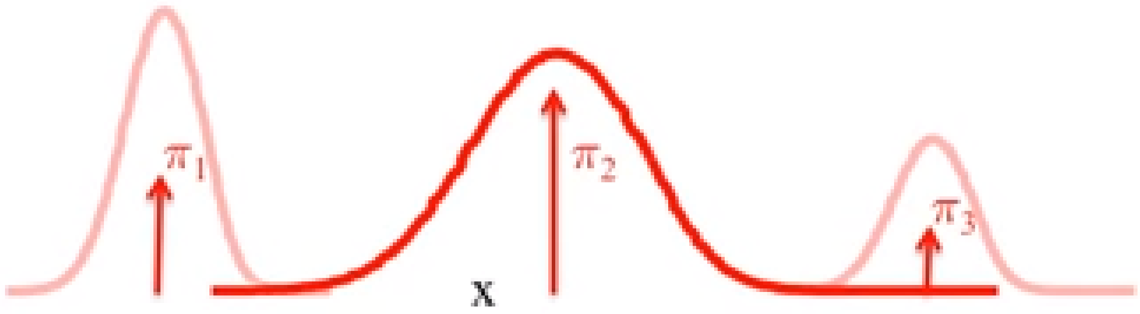
- When data is continuous, can be described by Normal Distributions



Gaussian Mixture Models

- Cluster data as Gaussians, with parameters: $(\mu_j, \sigma_j^2, \pi_j)$

$$p(z = j) = \pi_j$$
$$p(x|z = j) = N(x; \mu_i, \sigma_i^2)$$



EM algorithm in Gaussian Mixtures

Step 0) Initialize $\theta = \begin{Bmatrix} \mu_1, \dots, \mu_k \\ \sigma_1^2, \dots, \sigma_k^2 \\ \pi_1, \dots, \pi_k \end{Bmatrix}$ (assuming k clusters)

Step 1) Expectation: compute $r_{i,j}\ for\ each\ x_i$

$$r_{i,j} = \frac{\pi_{i,j}\, p(x|z = j)}{\sum_{n=1}^{k} \pi_{i,n}\, p(x|z = n)}$$

EM algorithm for Gaussian Mixture

Step 2) Maximization:

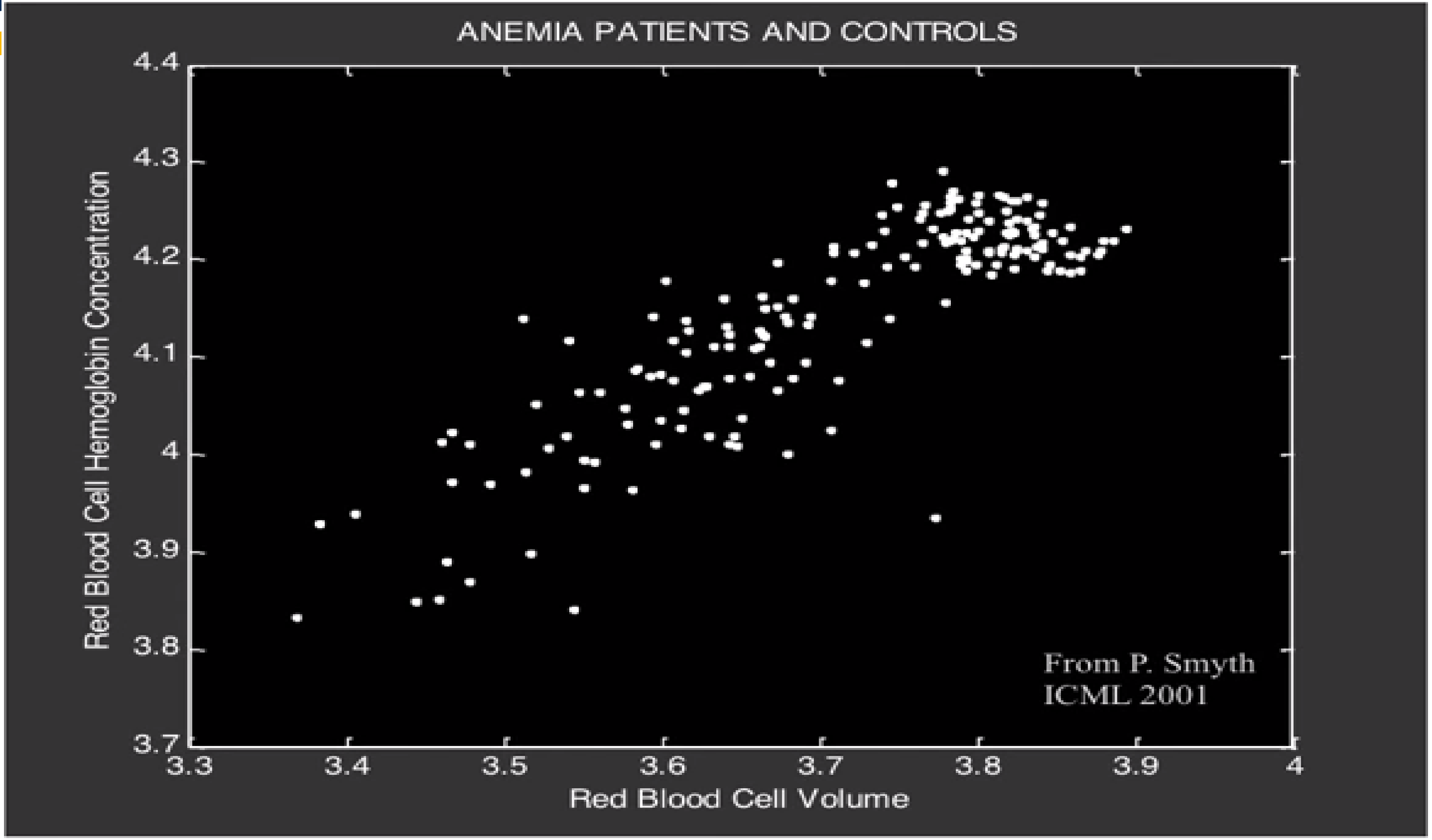
$$m_j = \sum_i r_{i,j}$$

$$\pi_j = \frac{m_j}{m}$$

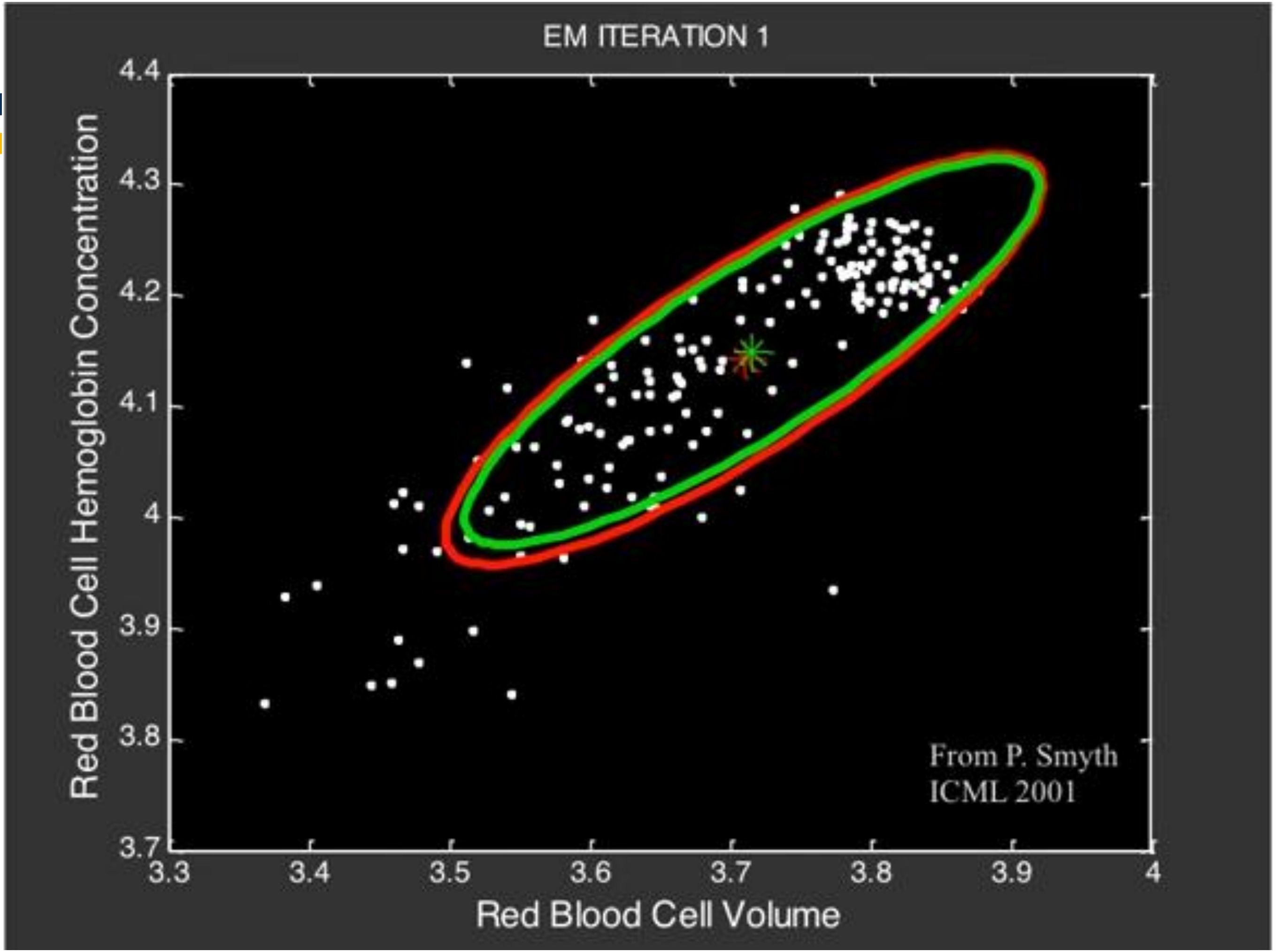
$$\mu_j = \frac{1}{m_j} \sum_i r_{i,j}\, x_i$$

$$\sigma_j^2 = \frac{1}{m_j} \sum_i r_{i,j} \left(x_i - \mu_j\right)^2$$

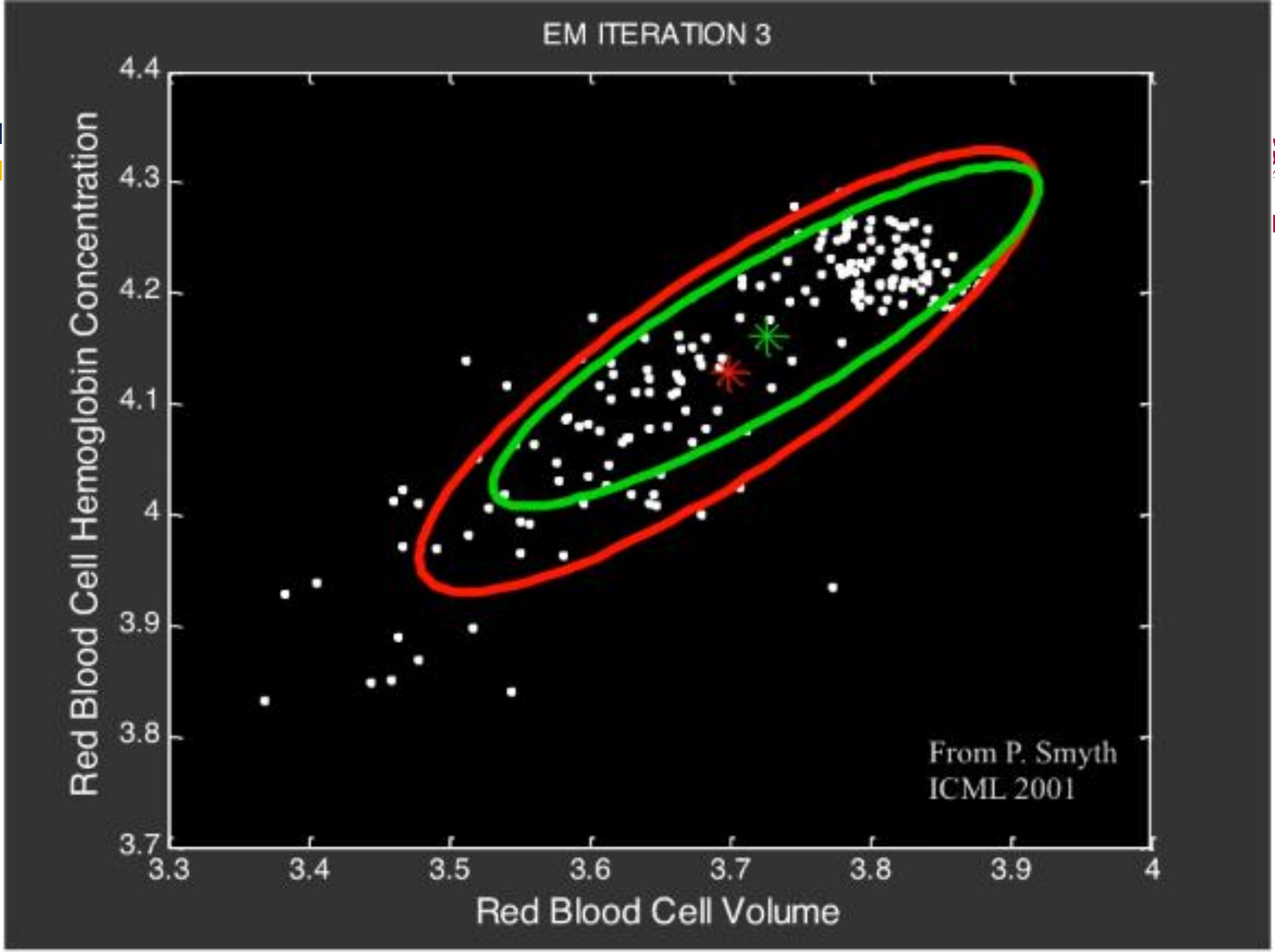
Example of EM in Gaussian Mixtures

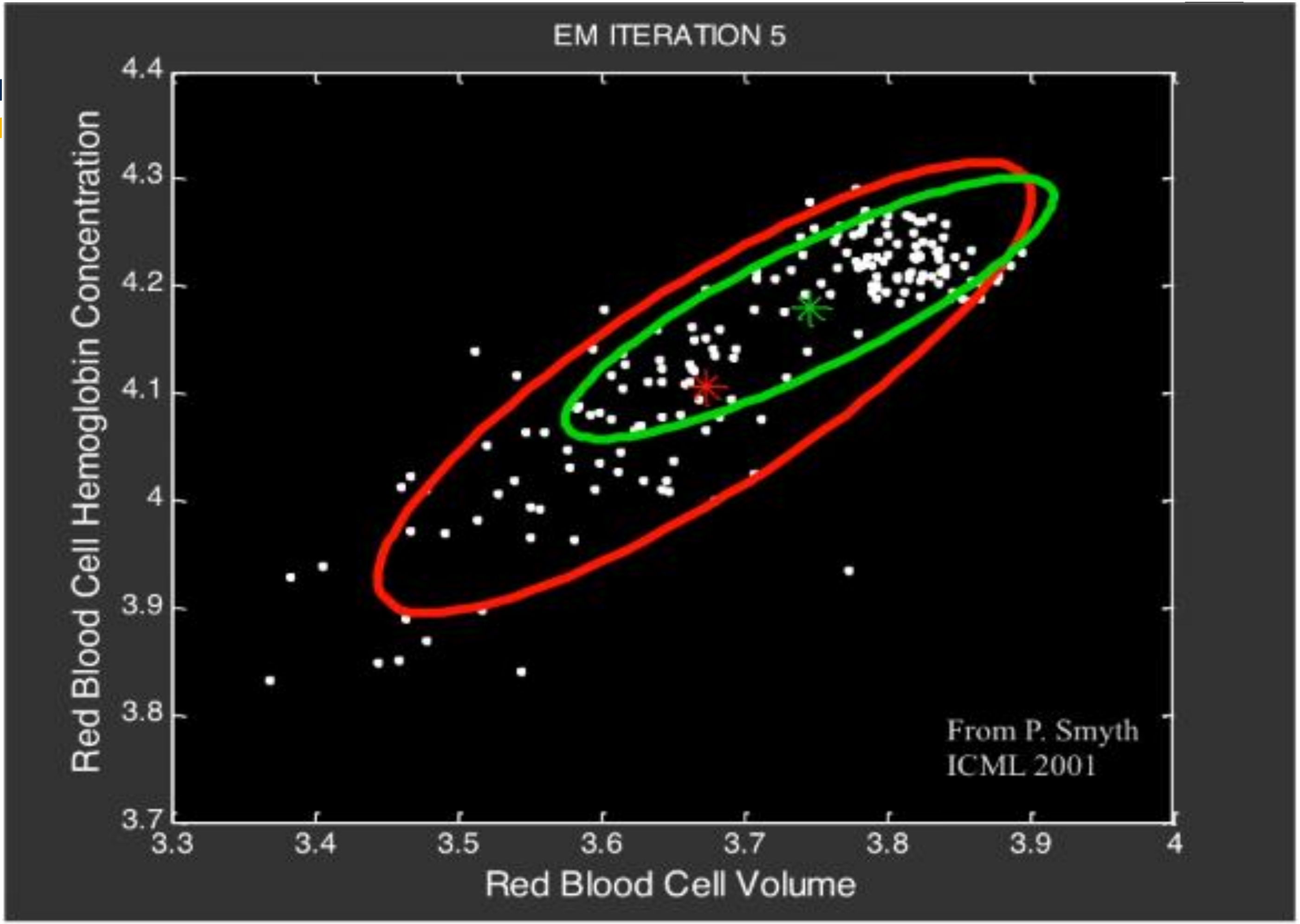


Alexander Ihler, https://www.youtube.com/watch?v=qMTuMa86NzU

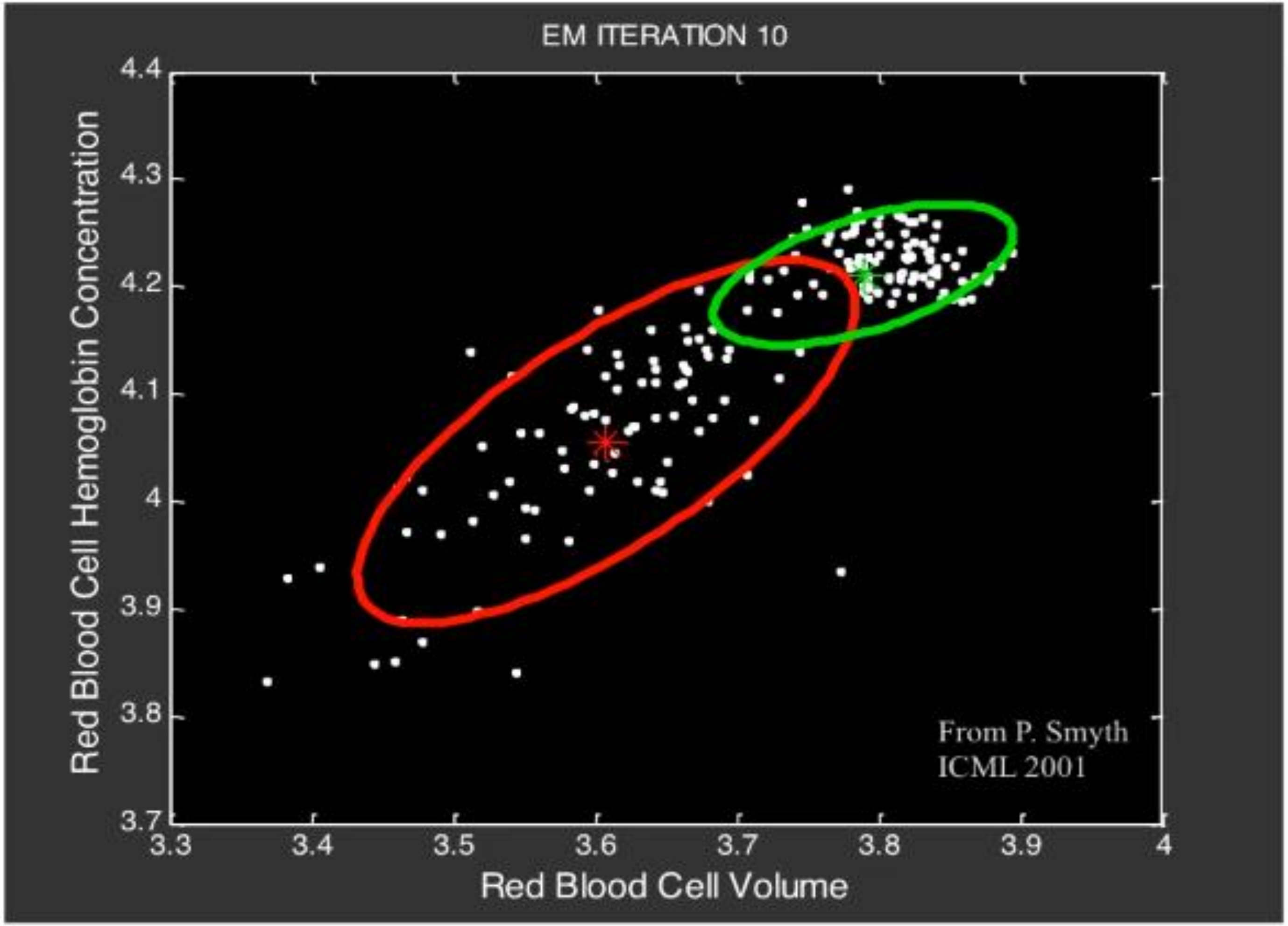


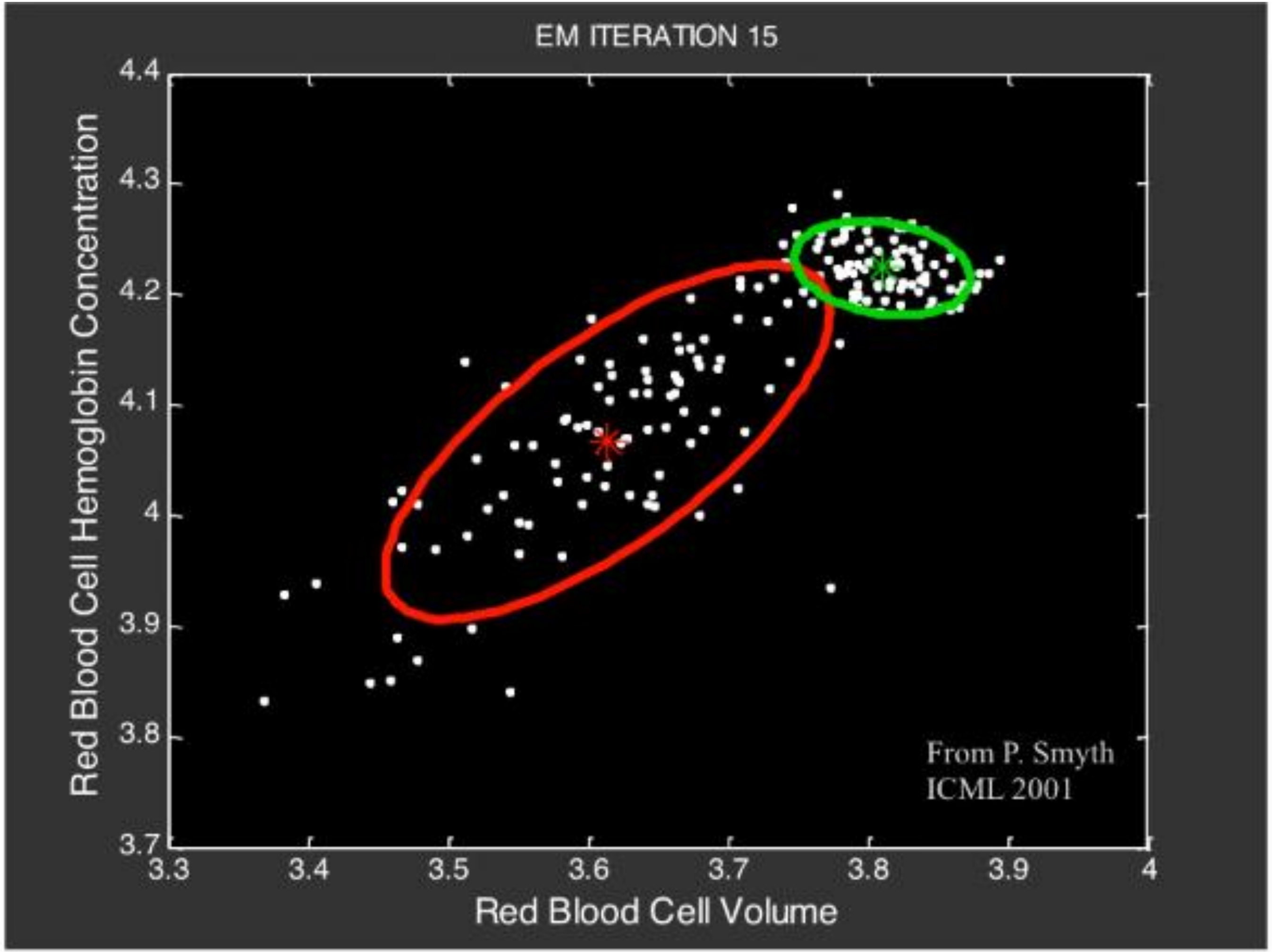
EM ITERATION 1
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001

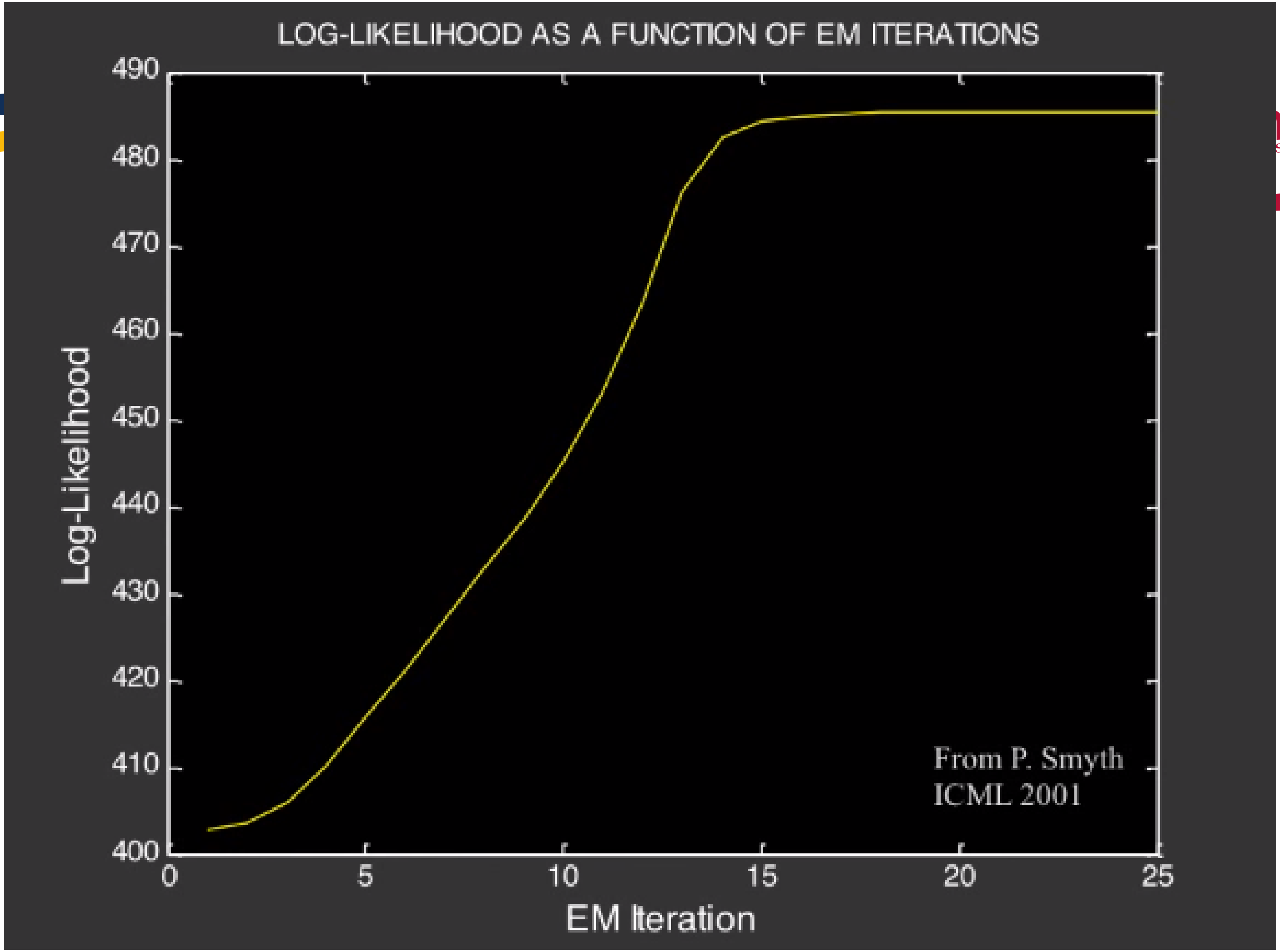

EM ITERATION 3
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001



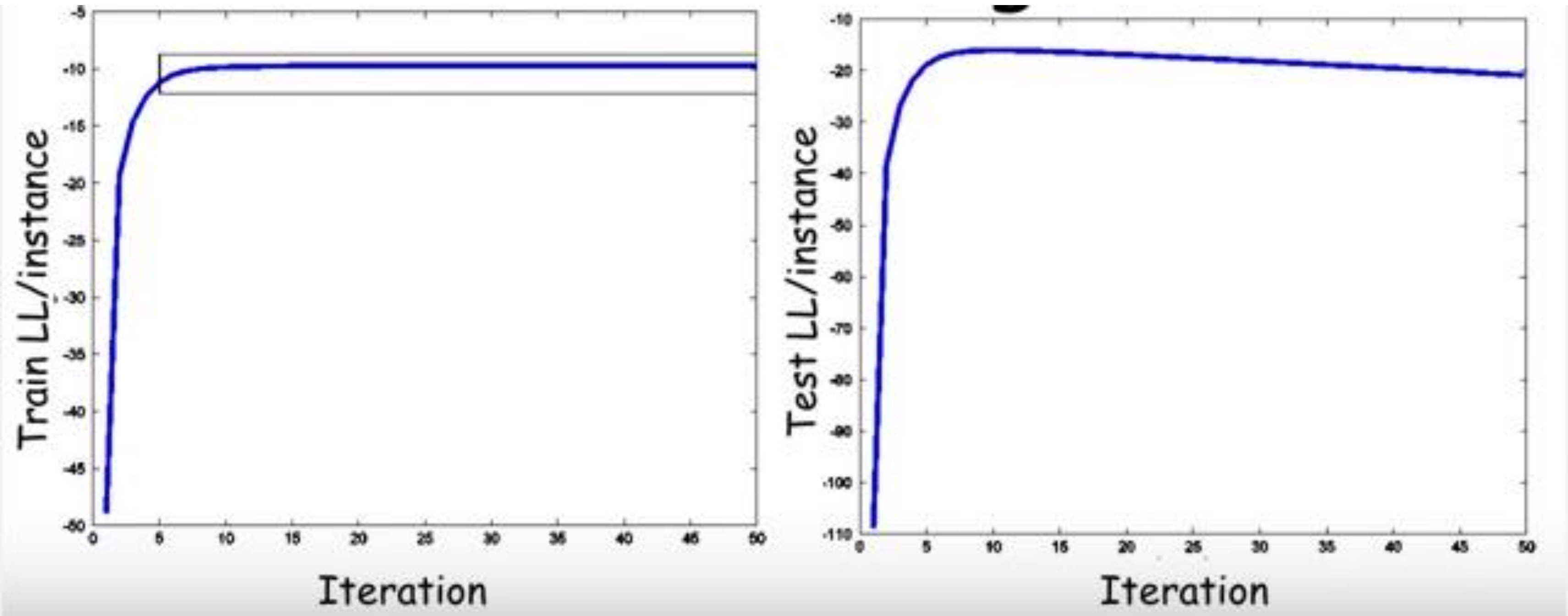
EM ITERATION 5
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4
From P. Smyth
ICML 2001


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


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

Overfitting through convergence



Initializing Parameters

- Hidden variables and incomplete data lead to more complex likelihood functions w/ many local optima
- Since EM only solves for a single local optima, choosing a good initial parameter estimation is critical
- Strategies to improve initialization
  - Multiple random restarts
  - Use prior knowledge
  - Output of a simpler, though less robust algorithm

Resources

- [Matlab EM Algorithm]
- Tom Mitchell- Machine Learning: Chapter 6 (on lab wiki)
- [EM Algorithm Derivation, Convergence, Hidden Markov and GMM Applications]
- [Nature Review Article]