CSE 473: Artificial Intelligence

Bayesian Networks - Learning

Dieter Fox

Slides adapted from Dan Weld, Jack Breese, Dan Klein, Daphne Koller, Stuart Russell, Andrew Moore & Luke Zettlemoyer

Space of ML Problems

Type of Supervision
(eg, Experience, Feedback)

What is Being Learned?

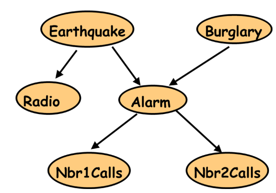
	Labeled Examples	Reward	Nothing
Discrete Function	Classification		Clustering
Continuous Function	Regression		
Policy	Apprenticeship Learning	Reinforcement Learning	

2

Learning Topics

- Learning Parameters for a Bayesian Network
  - Fully observable
    - Maximum Likelihood (ML)
    - Maximum A Posteriori (MAP)
    - Bayesian
  - Hidden variables (EM algorithm)
- Learning Structure of Bayesian Networks

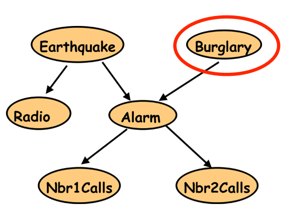
Parameter Estimation and Bayesian Networks



E	B	R	A	J	M
T	F	T	T	F	T
F	F	F	F	F	T
F	T	F	T	T	T
F	F	F	T	T	T
F	T	F	F	F	F
...					

We have:
- Bayes Net structure and observations
- We need: Bayes Net parameters

Parameter Estimation and Bayesian Networks

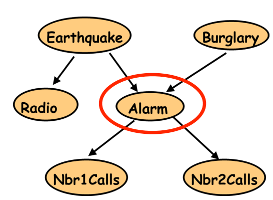


B
F
F
T
F
T

P(B) = ? = 0.4

P(¬B) = 1- P(B) = 0.6

Parameter Estimation and Bayesian Networks

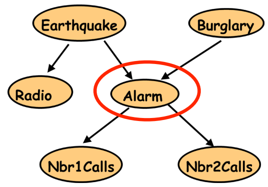


E	B
T	F
F	F
F	T
F	F
F	T
...	

A
T
F
T
T
F

P(A|E,B) = ?
P(A|E,¬B) = ?
P(A|¬E,B) = ?
P(A|¬E,¬B) = ?

Parameter Estimation and Bayesian Networks

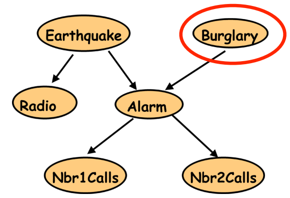


E	B
T	F
F	F
F	T
F	F
F	T
...	

A
T
F
T
T
F

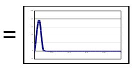
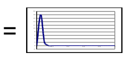
P(A|E,B) = ?
P(A|E,¬B) = ?
P(A|¬E,B) = ?
P(A|¬E,¬B) = 0.5

Parameter Estimation and Bayesian Networks



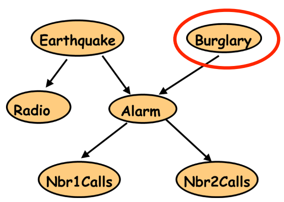
B
F
F
T
F
T

Prior

P(B) =  + data = 

Now compute either MAP or Bayesian estimate

Parameter Estimation and Bayesian Networks



B
F
F
T
F
T

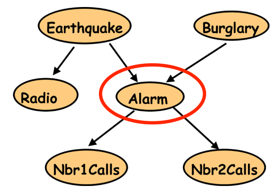
Prior

P(B|data) = **Beta(1,4)** "+ data" = **(3,7)**

B	¬B
.3	.7

Prior P(B)= 1/(1+4) = 20% with equivalent sample size 5

Parameter Estimation and Bayesian Networks

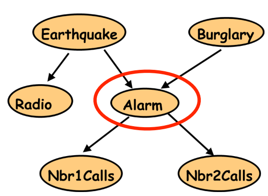


E	B
T	F
F	F
F	T
F	F
F	T
...	

A
T
F
T
T
F

P(A|E,B) = ?
P(A|E,¬B) = ?
P(A|¬E,B) = ?
P(A|¬E,¬B) = ?

Parameter Estimation and Bayesian Networks

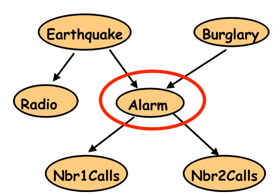


E	B
T	F
F	F
F	T
F	F
F	T
...	

A
T
F
T
T
F

P(A|E,B) = ? Prior
P(A|E,¬B) = ?
P(A|¬E,B) = ? **Beta(2,3)**
P(A|¬E,¬B) = ?

Parameter Estimation and Bayesian Networks



E	B
T	F
F	F
F	T
F	F
F	T
...	

A
T
F
T
T
F

P(A|E,B) = ? Prior
P(A|E,¬B) = ?
P(A|¬E,B) = ? **Beta(2,3)** + data= **(3,4)**
P(A|¬E,¬B) = ?

What if we *don't* know structure?

Learning The Structure of Bayesian Networks

- Search through the space…
  - of possible network structures!
  - (for now, assume we observe all variables)
- For each structure, learn parameters
- Pick the one that fits observed data best
  - Caveat – won't we end up fully connected????

When scoring, add a penalty
model complexity

Learning The Structure of Bayesian Networks

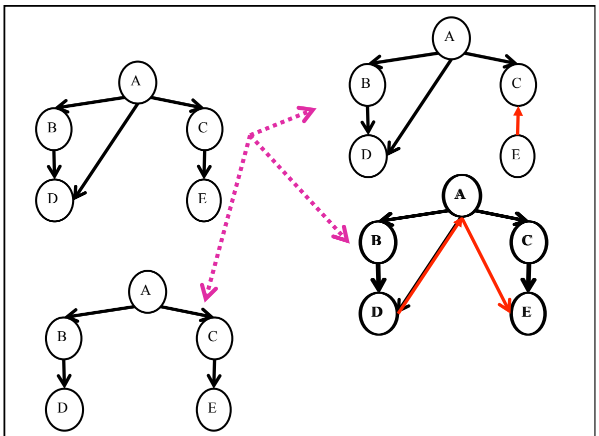
- Search through the space
- For each structure, learn parameters
- Pick the one that fits observed data best
  - Penalize complex models
- Problem?

Exponential number of networks!
And we need to learn parameters for each!
Exhaustive search out of the question!

Structure Learning as Search

- Local Search
1. Start with some network structure
2. Try to make a change
   (add or delete or reverse edge)
3. See if the new network is any better
- What should the initial state be?
  - Uniform prior over random networks?
  - Based on prior knowledge?
  - Empty network?
- How do we evaluate networks?

16



Scoring a Bayes Net Structure

- Bayesian Information Criterion (BIC)
  - P(D | BN) – penalty
  - Penalty = ½ (# parameters) Log (# data points)

18

Expectation Maximization and Gaussian Mixtures

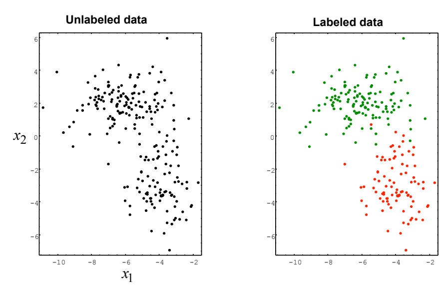
CSE 473

Feedback in Learning

- Supervised learning: correct answers for each example
- Unsupervised learning: correct answers not given
- Reinforcement learning: occasional rewards

The problem of finding labels for unlabeled data

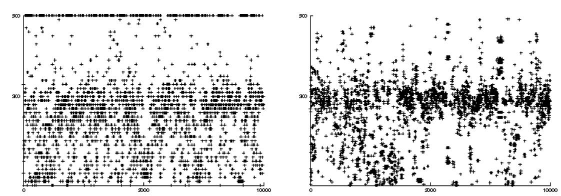
In nature, items often do not come with labels. How can we learn labels without a teacher?



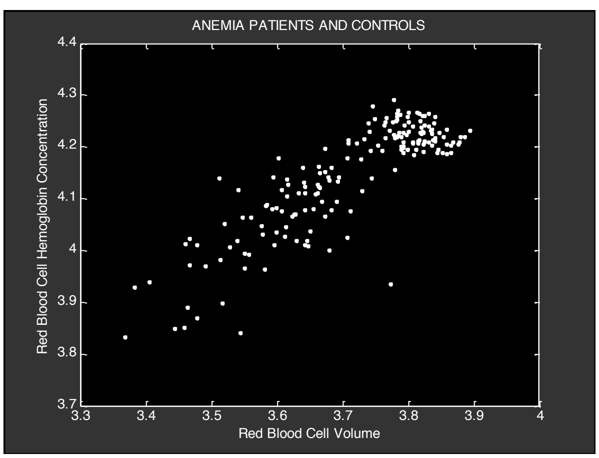
From Shadmehr & Diedrichsen

Raw Proximity Sensor Data

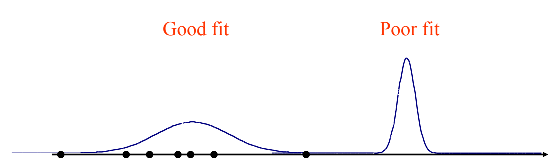
Measured distances for expected distance of 300 cm.



Sonar Laser



Fitting a Gaussian PDF to Data



Fitting a Gaussian PDF to Data

- Suppose $y = y_1,\ldots,y_n,\ldots,y_N$ is a set of N data values
- Given a Gaussian PDF p with mean μ and std dev σ, define:

$$p(y\mid\mu,\sigma) = \prod_{n=1}^{N} p(y_n\mid\mu,\sigma) = \prod_{n=1}^{N} \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{1}{2}\frac{(y_n-\mu)^2}{\sigma^2}}$$

- How do we choose μ and σ to maximise this probability?

Fisher, 1922

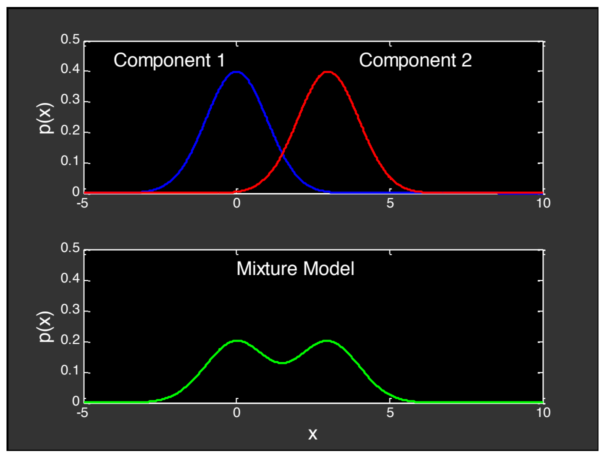
Maximum Likelihood Estimation

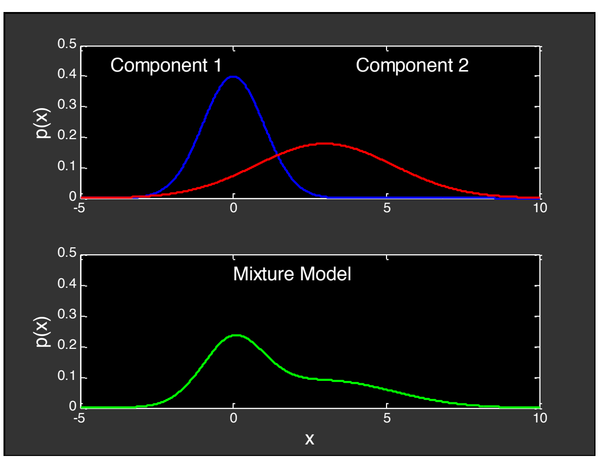
- Define the best fitting Gaussian to be the one such that $p(y|\mu,\sigma)$ is maximized.
- Terminology:
  - $p(y|\mu,\sigma)$, thought of as a function of y is the probability (density) of y
  - $p(y|\mu,\sigma)$, thought of as a function of μ, σ is the likelihood of μ, σ
- Maximizing $p(y|\mu,\sigma)$ with respect to μ,σ is called Maximum Likelihood (ML) estimation of μ, σ

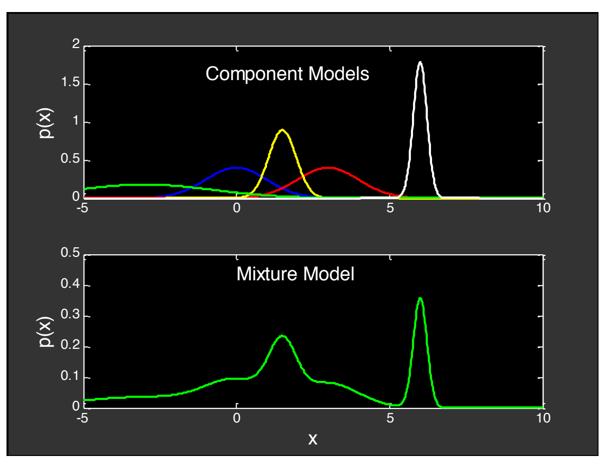
ML estimation of μ,σ

- Intuitively:
  - The maximum likelihood estimate of μ should be the average value of $y_1,\ldots,y_N$, (the sample mean)
  - The maximum likelihood estimate of σ should be the variance of $y_1,\ldots,y_N$. (the sample variance)
- This turns out to be true: $p(y|\mu,\sigma)$ is maximized by setting:

$$\mu = \frac{1}{N}\sum_{n=1}^{N} y_n, \qquad \sigma = \frac{1}{N}\sum_{n=1}^{N} (y_n-\mu)^2$$







Mixtures

If our data is not labeled, we can hypothesize that:

1. There are exactly m classes in the data: $y \in \{1, 2, \ldots, m\}$
2. Each class y occurs with a specific frequency: $P(y)$
3. Examples of class y are governed by a specific distribution: $p(\mathbf{x}|y)$

According to our hypothesis, each example $\mathbf{x}^{(i)}$ must have been generated from a specific "mixture" distribution:

$$p(\mathbf{x}) = \sum_{j=1}^{m} P(y=j)\, p(\mathbf{x}|y=j)$$

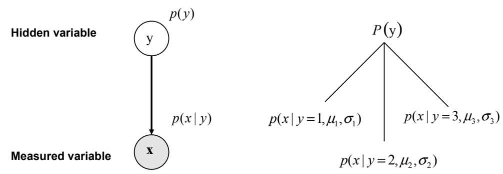
We might hypothesize that the distributions are Gaussian:

Parameters of the distributions $\theta = \{P(y=1), \mu_1, \Sigma_1, \cdots, P(y=m), \mu_m, \Sigma_m\}$

$$p(\mathbf{x}|\theta) = \sum_{j=1}^{m} P(y=j)\, N(\mathbf{x}|\mu_j, \Sigma_j)$$

Mixing proportions **Normal distribution**

Graphical Representation of Gaussian Mixtures



$$p(x) = \sum_{i=1}^{3} p(y=i)\, p(x \mid y=i, \mu_i, \sigma_i)$$

Learning of mixture models

Learning Mixtures from Data

Consider fixed K = 2

e.g., unknown parameters Q = {m_1, s_1, m_2, s_2, a_l}

Given data D = {x_1,…….x_N}, we want to find the parameters Q that "best fit" the data

1977: The EM Algorithm

- Dempster, Laird, and Rubin
  - General framework for likelihood-based parameter estimation with missing data
    - start with initial guesses of parameters
    - E-step: estimate memberships given params
    - M-step: estimate params given memberships
    - Repeat until convergence
  - Converges to a **local** maximum of likelihood
  - E-step and M-step are often computationally simple
  - Can incorporate priors over parameters

EM for Mixture of Gaussians

- E-step: Compute probability that point x_j was generated by component i:

$$p_{ij} = \alpha\, P(x_j \mid C=i)\, P(C=i)$$

$$p_i = \sum_j p_{ij}$$

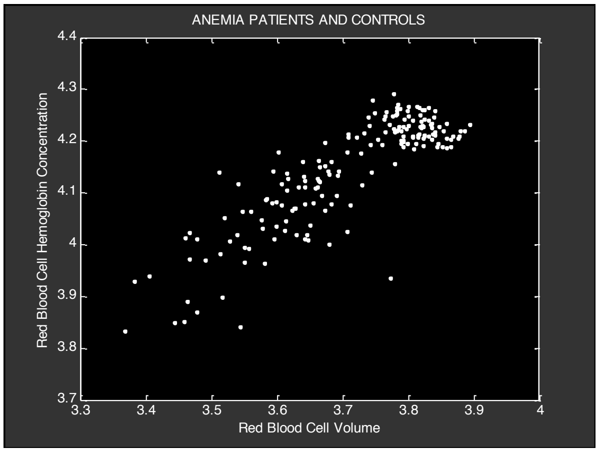
- M-step: Compute new mean, covariance, and component weights:

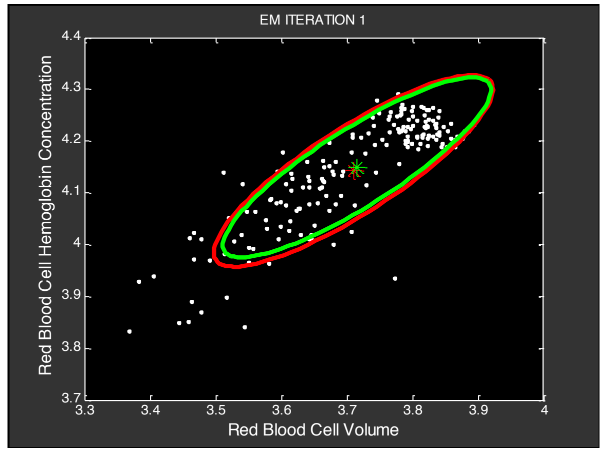
$$\mu_i \leftarrow \sum_j p_{ij} x_j / p_i$$

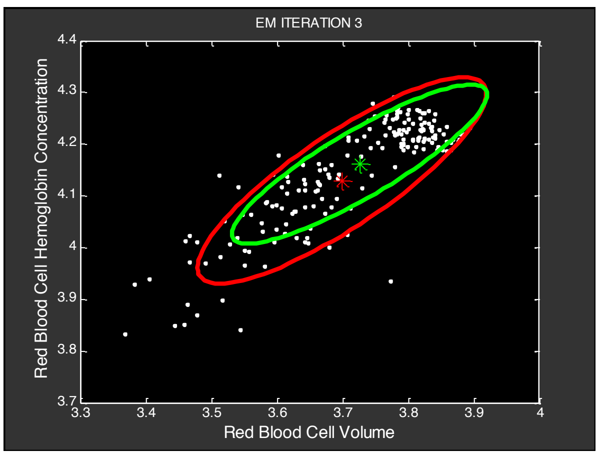
$$\sigma^2 \leftarrow \sum_j p_{ij} (x_j - \mu_i)^2 / p_i$$

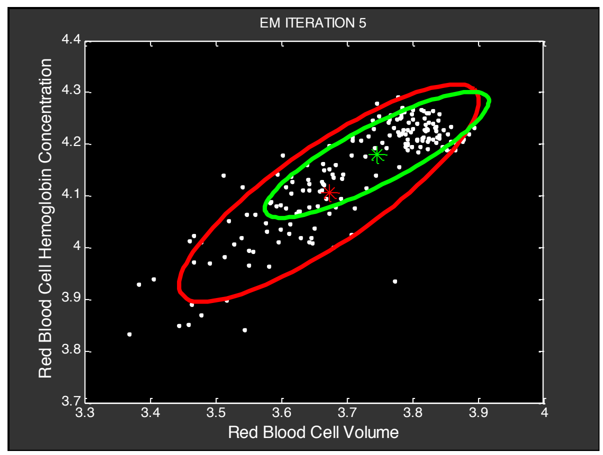
$$w_i \leftarrow p_i$$

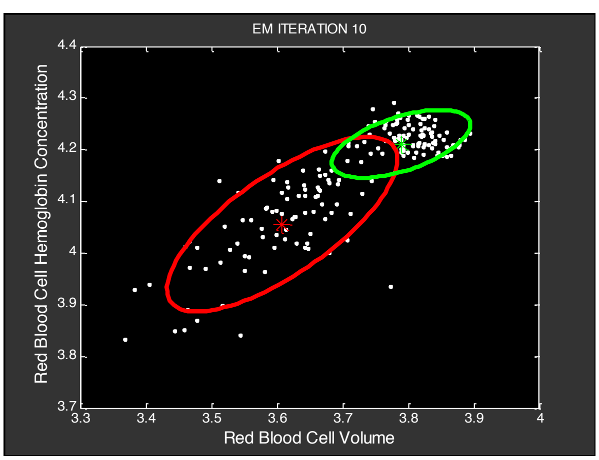
36

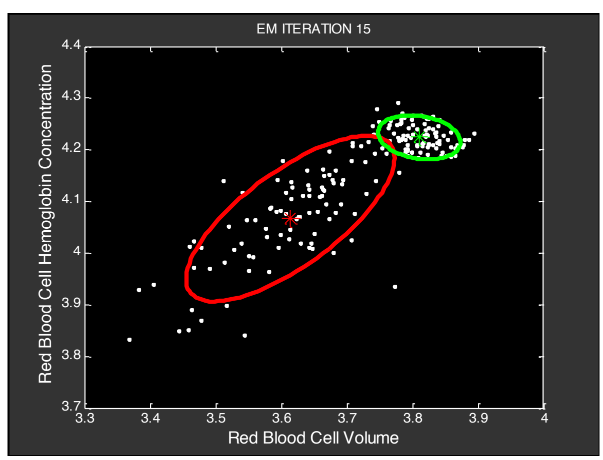

ANEMIA PATIENTS AND CONTROLS
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume
4.4
4.3
4.2
4.1
4
3.9
3.8
3.7
3.3
3.4
3.5
3.6
3.7
3.8
3.9
4

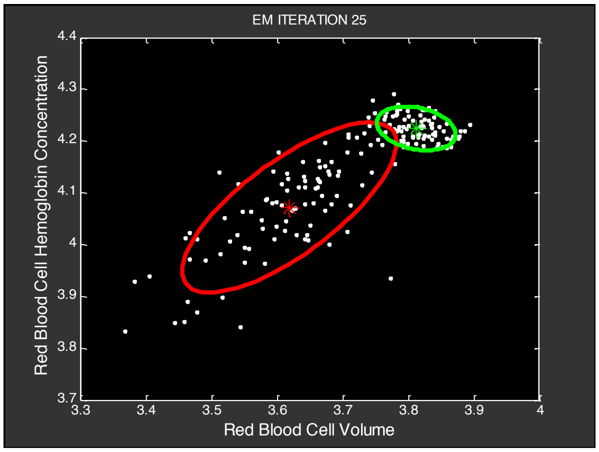

EM ITERATION 1
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume

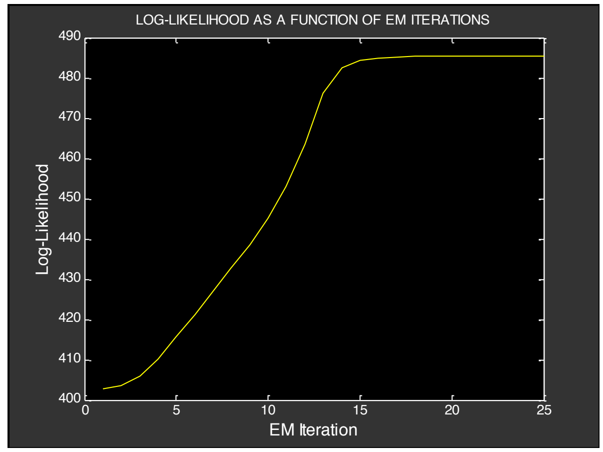

EM ITERATION 3
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume

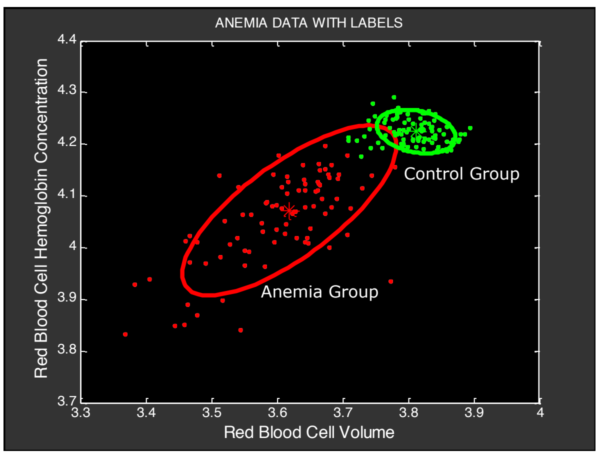

EM ITERATION 5
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume


EM ITERATION 10
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume

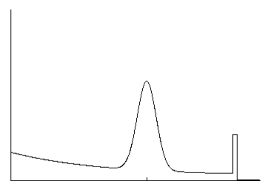

EM ITERATION 15
Red Blood Cell Hemoglobin Concentration
Red Blood Cell Volume







Mixture Density

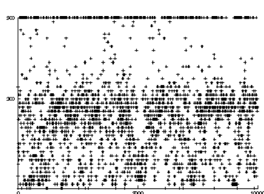


$$P(z \mid x, m) = \begin{pmatrix} \alpha_{\text{hit}} \\ \alpha_{\text{unexp}} \\ \alpha_{\max} \\ \alpha_{\text{rand}} \end{pmatrix}^T \cdot \begin{pmatrix} P_{\text{hit}}(z \mid x, m) \\ P_{\text{unexp}}(z \mid x, m) \\ P_{\max}(z \mid x, m) \\ P_{\text{rand}}(z \mid x, m) \end{pmatrix}$$

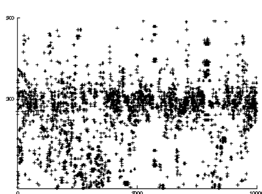
How can we determine the model parameters?

Raw Sensor Data

Measured distances for expected distance of 300 cm.

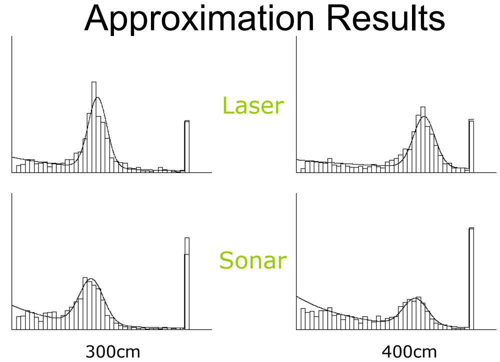


Sonar

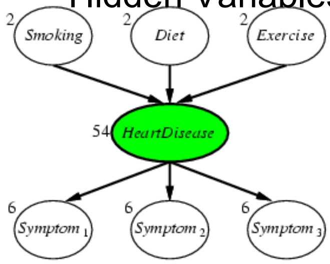


Laser

Approximation Results



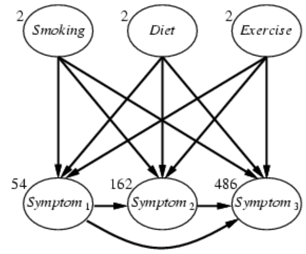
Hidden Variables



- But we can't observe the disease variable
- Can't we learn without it?

We *–could-*

- But we'd get a fully-connected network

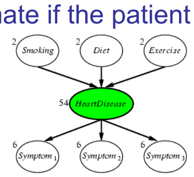


With 708 parameters (vs. 78)

Much harder to learn!

Chicken & Egg Problem

- If we knew that a training instance (patient) had the disease, then it'd be easy to learn P(symptom | disease)
- If we knew params, e.g. P(symptom | disease) then it'd be easy to estimate if the patient had the disease



Expectation Maximization (EM)

(high-level version)

- Pretend we ***do*** know the parameters
  - Initialize randomly
- **[E step]** Compute probability of instance having each possible value of the hidden variable
- **[M step]** Treating each instance as fractionally having both values compute the new parameter values
- Iterate until convergence!

52