EECS 445

Introduction to Machine Learning

EM Algorithm and Bayesian Networks

Prof. Kutty

Announcements

Course evaluations are out

- Gradescope assignment to upload proof (screenshot)
  - **Please note *separate*** eval and assignment deadlines!!!
    - deadline for the assignment is *different* from the registrar's deadline
- worth 0.5% of your grade!

HW4 due in one week: please get started early!

Special Topics Course next term:

EECS498 (Machine Learning Research Experience)

EECS498 (Algorithms for Data Science)

MICHIGAN INSTITUTE FOR DATA SCIENCE
UNIVERSITY OF MICHIGAN

Data Science Night @ U-M

April 19th, 4:30 - 8:30 PM @ CCCB

Interested in data science @ U-M? Join us for a night of presentations, raffles, simulations and more! Connect with top data science student organizations and hear about their project work in data analysis, machine learning, AI, and more and learn about ways to get involved!

Open to all U-M Faculty and Students. Food will be provided.

Event Schedule

- **Introduction + Project Presentations**
  4:30-5:45pm @ CCCB 0420
- **Digital Poster Session**
  5:45-6:30pm @ CCCB 0460
- **AI Ethics Simulation**
  6:30-7:30pm @ CCCB 0460
- **Project Presentations**
  7:45-8:15pm @ CCCB 0420
- **Wrap Up + Awards**
  7:45-8:15pm @ CCCB 0420

RSVP

tinyurl.com/ds-night-um

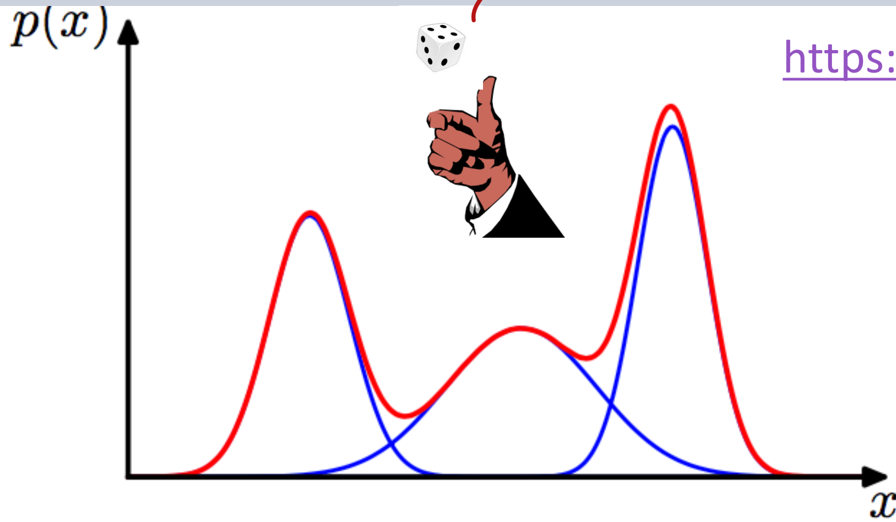
Michigan Data Science Team

MAISI

stat.com

Mixture of Gaussians

k-sided die } eg, k=3



https://forms.gle/ffiBvNbPjHF8ghi7

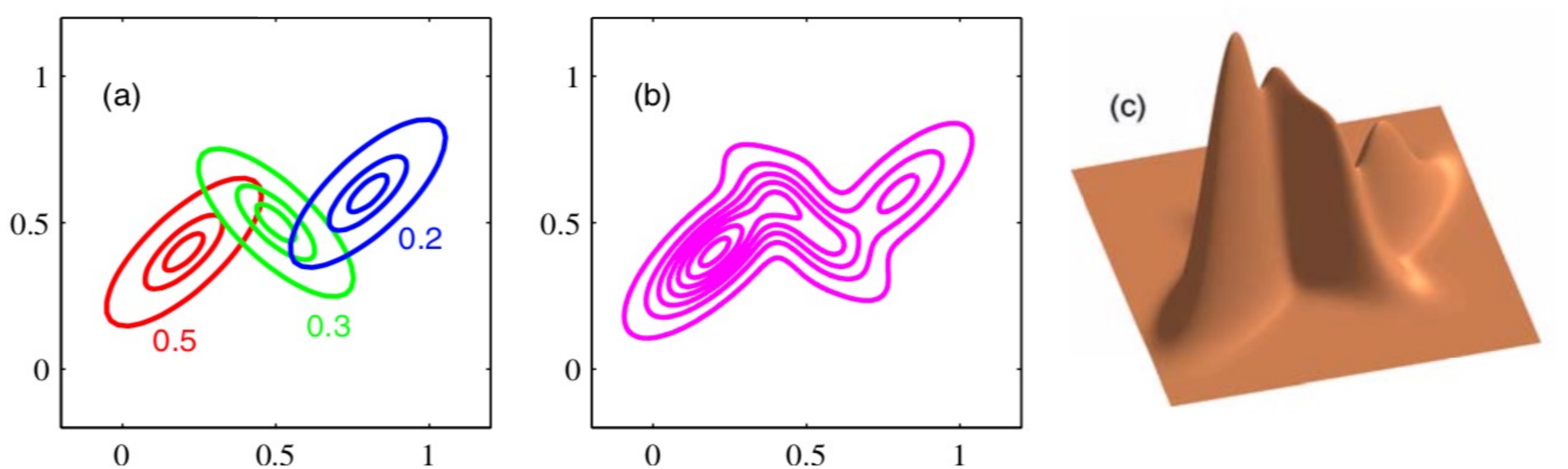
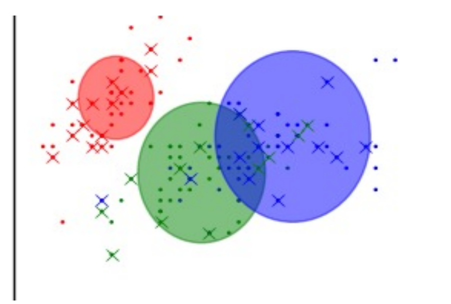


Figure 2.23 Illustration of a mixture of 3 Gaussians in a two-dimensional space. (a) Contours of constant density for each of the mixture components, in which the 3 components are denoted red, blue and green, and the values of the mixing coefficients are shown below each component. (b) Contours of the marginal probability density $p(\mathbf{x})$ of the mixture distribution. (c) A surface plot of the distribution $p(\mathbf{x})$.

image source: Bishop 2006

MLE for GMMs with known labels



Likelihood for Mixture Models with known labels

$$P(S_n) = \prod_{i=1}^{n} p(\underbrace{\bar{x}^{(i)}}_{\in \mathbb{R}^d}, \underbrace{y^{(i)}}_{\in \{1,\dots,k\}})$$

$$= \prod_{i=1}^{n} p(\bar{x}^{(i)} | y^{(i)}) p(y^{(i)})$$

Recall from probability

product rule:
$$p(A, B) = p(A|B)p(B)$$

sum rule:
$$p(A) = \sum_{B} p(A, B)$$

Likelihood for GMMs with known labels

pdf: probability density function

$p(y=c_1) = \gamma_1$

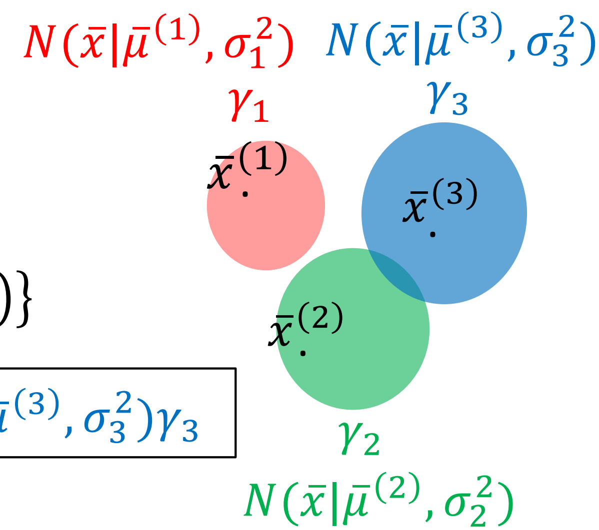
$p(\bar{x}^{(1)} = \begin{bmatrix} 3 \\ 5 \end{bmatrix} \mid y=c_1)$

$N(\bar{x}^{(1)} \mid \bar{\mu}^{(1)}, \sigma_1^2)$

$$P(S_n) = \prod_{i=1}^{n} p(\bar{x}^{(i)}, y^{(i)})$$

$$= \prod_{i=1}^{n} p(\bar{x}^{(i)} | y^{(i)}) p(y^{(i)})$$

$$= \prod_{i=1}^{n} \sum_{j=1}^{k} \delta(j \mid i)(N(\bar{x}^{(i)} | \bar{\mu}^{(j)}, \sigma_j^2)\gamma_j)$$



<u>example</u>: $S_n = \{(\bar{x}^{(1)}, c_1), (\bar{x}^{(2)}, c_2), (\bar{x}^{(3)}, c_3)\}$

Likelihood $\boxed{N(\bar{x}^{(1)} | \bar{\mu}^{(1)}, \sigma_1^2)\gamma_1 \times N(\bar{x}^{(2)} | \bar{\mu}^{(2)}, \sigma_2^2)\gamma_2 \times N(\bar{x}^{(3)} | \bar{\mu}^{(3)}, \sigma_3^2)\gamma_3}$

Define indicator function

$$\delta(j \mid i) = \begin{cases} 1 \text{ if } \bar{x}^{(i)} \text{ belongs to cluster } j \\ 0 \quad \text{otherwise} \end{cases}$$

$\delta(1|1) = 1$

$\delta(2|1) = 0$

$\delta(3|1) = 0$

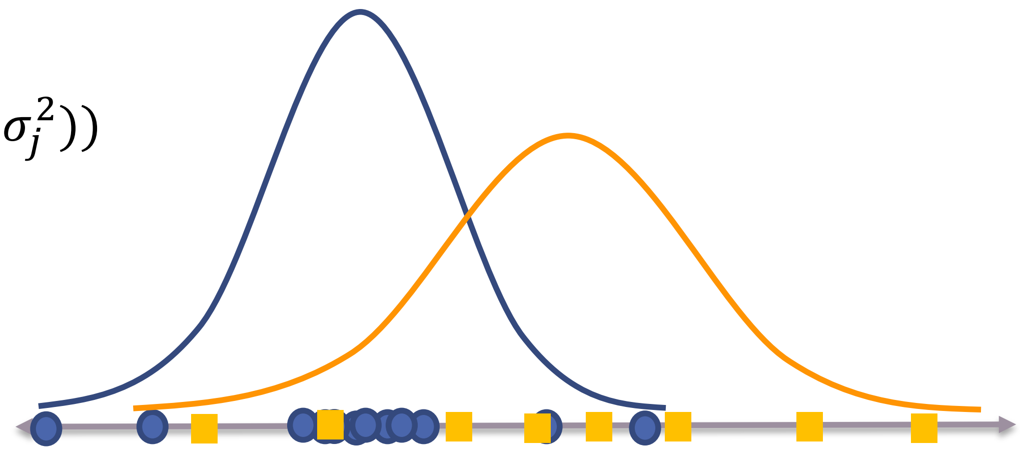
Note that

$$\sum_{j=1}^{3} \delta(j \mid 1)(N(\bar{x}^{(1)} | \bar{\mu}^{(J)}, \sigma_j^2)\gamma_j) = N(\bar{x}^{(1)} | \bar{\mu}^{(1)}, \sigma_1^2)\gamma_1$$

MLE for GMMs with known labels

Maximum log likelihood objective

$$\sum_{i=1}^{n}\sum_{j=1}^{k}\delta(j \mid i)\ln\left(\gamma_j N(\bar{x}|\bar{\mu}^{(j)},\sigma_j^2)\right)$$



MLE solution (given "cluster labels"):

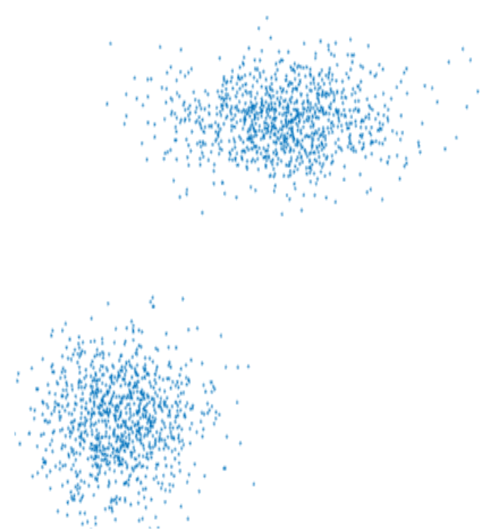
Define $\hat{n}_j = \sum_{i=1}^{n}\delta(j \mid i)$ number of points assigned to cluster j

$$\gamma_j = \frac{\hat{n}_j}{n}$$ fraction of points assigned to cluster j

$$\bar{\mu}^{(j)} = \frac{1}{\hat{n}_j}\sum_{i=1}^{n}\delta(j \mid i)\ \bar{x}^{(i)}$$ mean of points in cluster j

$$\sigma_j^2 = \frac{1}{d\hat{n}_j}\sum_{i=1}^{n}\delta(j \mid i)\left\|\bar{x}^{(i)} - \bar{\mu}^{(j)}\right\|^2$$ spread in cluster j

MLE for GMMs with *un*known labels



Likelihood for Mixture Models with *un*known labels

$$P(S_n) = \prod_{i=1}^{n} p(\bar{x}^{(i)}) = \prod_{i=1}^{n} \sum_{y^{(i)} \in \{1,\ldots,k\}} p(\bar{x}^{(i)}, y^{(i)})$$

$$= \prod_{i=1}^{n} \sum_{y^{(i)} \in \{1,\ldots,k\}} p(\bar{x}^{(i)} | y^{(i)}) p(y^{(i)})$$

Recall from probability

product rule:
$p(A, B) \ = p(A|B)p(B)$

sum rule:

$$p(A) \ = \sum_{B} p(A, B)$$

marginalizing out B (discrete r.v.)

$\hookrightarrow$ random variable

Learning the Model Parameters

$$P(S_n) = \prod_{i=1}^{n} p(\bar{x}^{(i)}) = \prod_{i=1}^{n} \sum_{j=1}^{k} p(\bar{x}^{(i)}, y^{(i)} = j)$$

$$= \prod_{i=1}^{n} \sum_{j=1}^{k} p(\bar{x}^{(i)} | y^{(i)} = j) p(y^{(i)} = j)$$

$$= \prod_{i=1}^{n} \sum_{j=1}^{k} N(\bar{x}^{(i)} | \bar{\mu}^{(j)}, \sigma_j^2) \gamma_j$$

Given the training data, find the model parameters that maximize the **log-likelihood**

$$\ln(P(S_n))$$

$$= \ln\left(\prod_{i=1}^{n} \sum_{j=1}^{k} N(\bar{x}^{(i)} | \bar{\mu}^{(j)}, \sigma_j^2) \gamma_j\right) = \sum_{i=1}^{n} \ln\left(\sum_{j=1}^{k} N(\bar{x}^{(i)} | \bar{\mu}^{(j)}, \sigma_j^2) \gamma_j\right)$$

model parameters: $\bar{\theta} = [\overbrace{\gamma_1, \ldots, \gamma_k}^{k-1}, \overbrace{\bar{\mu}^{(1)}, \ldots, \bar{\mu}^{(k)}}^{kd}, \overbrace{\sigma_1^2, \ldots, \sigma_k^2}^{k}]$

$\bar{x}^{(i)} \in \mathbb{R}^d$

where each mixture component is a spherical Gaussian and γ_j are the mixing coefficients

Expectation Maximization for GMMs

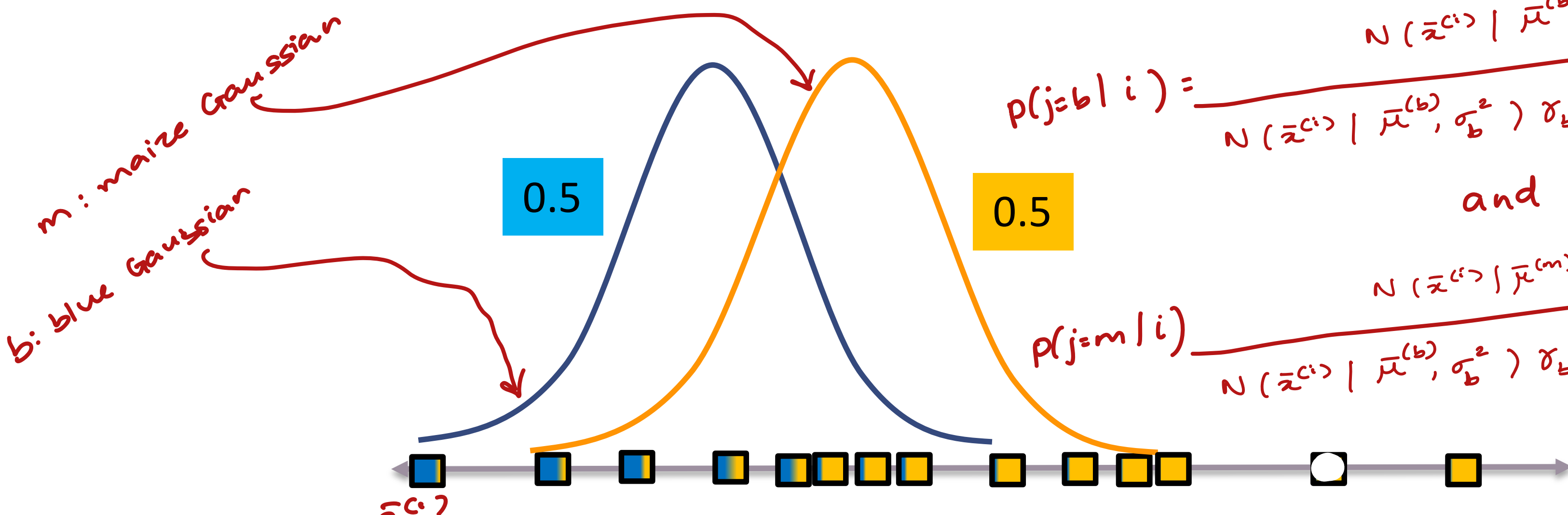
Expectation Maximization for GMMs

- **E-step**:

fix $\bar{\theta} = [\gamma_1, \ldots, \gamma_k, \bar{\mu}^{(1)}, \ldots, \bar{\mu}^{(k)}, \sigma_1^2, \ldots, \sigma_k^2]$

softly assign points to clusters according to posterior prob

$$p(j|i) = \frac{\gamma_j N(\bar{x}^{(i)} \mid \bar{\mu}^{(j)}, \sigma_j^2)}{\sum_t \gamma_t N(\bar{x}^{(i)} \mid \bar{\mu}^{(t)}, \sigma_t^2)}$$



$$p(j=b \mid i) = \frac{N(\bar{x}^{(i)} \mid \bar{\mu}^{(b)}, \sigma_b^2)\,\gamma_b}{N(\bar{x}^{(i)} \mid \bar{\mu}^{(b)}, \sigma_b^2)\,\gamma_b + N(\bar{x}^{(i)} \mid \bar{\mu}^{(m)}, \sigma_m^2)\,\gamma_m}$$

and

$$p(j=m \mid i) = \frac{N(\bar{x}^{(i)} \mid \bar{\mu}^{(m)}, \sigma_m^2)\,\gamma_m}{N(\bar{x}^{(i)} \mid \bar{\mu}^{(b)}, \sigma_b^2)\,\gamma_b + N(\bar{x}^{(i)} \mid \bar{\mu}^{(m)}, \sigma_m^2)\,\gamma_m}$$

"soft" cluster assignment $p(j|i)$
given a datapoint $\bar{x}^{(i)}$ what is the probability that cluster j generated it
Analogous to $\delta(j|i)$ note that $\sum_j p(j|i) = 1$

and

Expectation Maximization for GMMs
E-step: Example

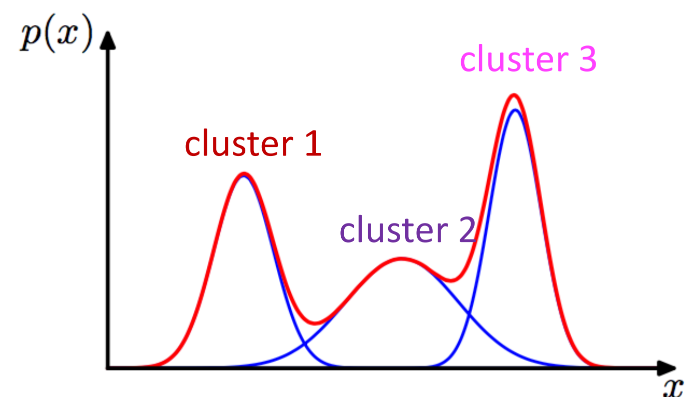
E-step: softly assign points to clusters according to current guess of model parameters

$$p(j|i) = \frac{\gamma_j P(\bar{x}^{(i)}|\bar{\mu}^{(j)}, \sigma_j^2)}{P(\bar{x}^{(i)}|\bar{\theta})}$$

Example

Datapoints	Cluster 1 0.5	Cluster 2 0.3	Cluster 3 0.2
$x^{(1)} = 0$	mean = 0 variance = 1	mean = 1 variance = 1	mean = 3 variance = 4
$x^{(2)} = 1$			
$x^{(3)} = 3$			
$x^{(4)} = 2$			
$x^{(5)} = 5$			

which is the likeliest cluster for datapoint $x^{(1)}$



$$P\left(x^{(1)}|\mu^{(1)}, \sigma_1^2\right) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left[-\frac{\left(x^{(1)} - \mu^{(1)}\right)^2}{2\sigma_1^2}\right]$$

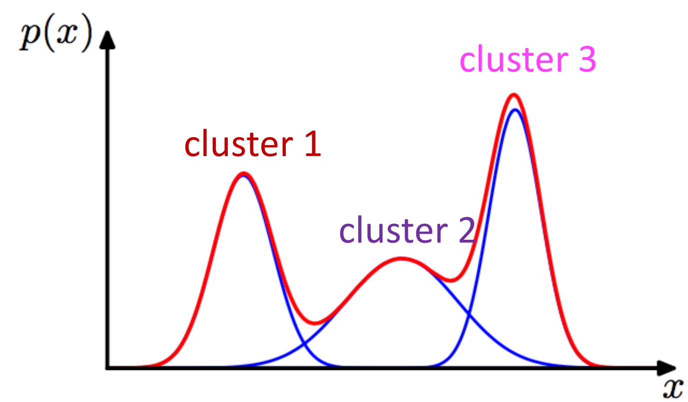
Expectation Maximization for GMMs
E-step: Example

E-step: softly assign points to clusters according to current guess of model parameters

$$p(j|i) = \frac{\gamma_j P(\bar{x}^{(i)}|\bar{\mu}^{(j)}, \sigma_j^2)}{P(\bar{x}^{(i)}|\bar{\theta})}$$

Example

Datapoints	Cluster 1 0.5	Cluster 2 0.3	Cluster 3 0.2
$x^{(1)} = 0$	mean = 0 variance = 1	mean = 1 variance = 1	mean = 3 variance = 4
$x^{(2)} = 1$			
$x^{(3)} = 3$			
$x^{(4)} = 2$			
$x^{(5)} = 5$			



0.5 * 0.39894

0.2 * 0.06476

0.3 * 0.24197

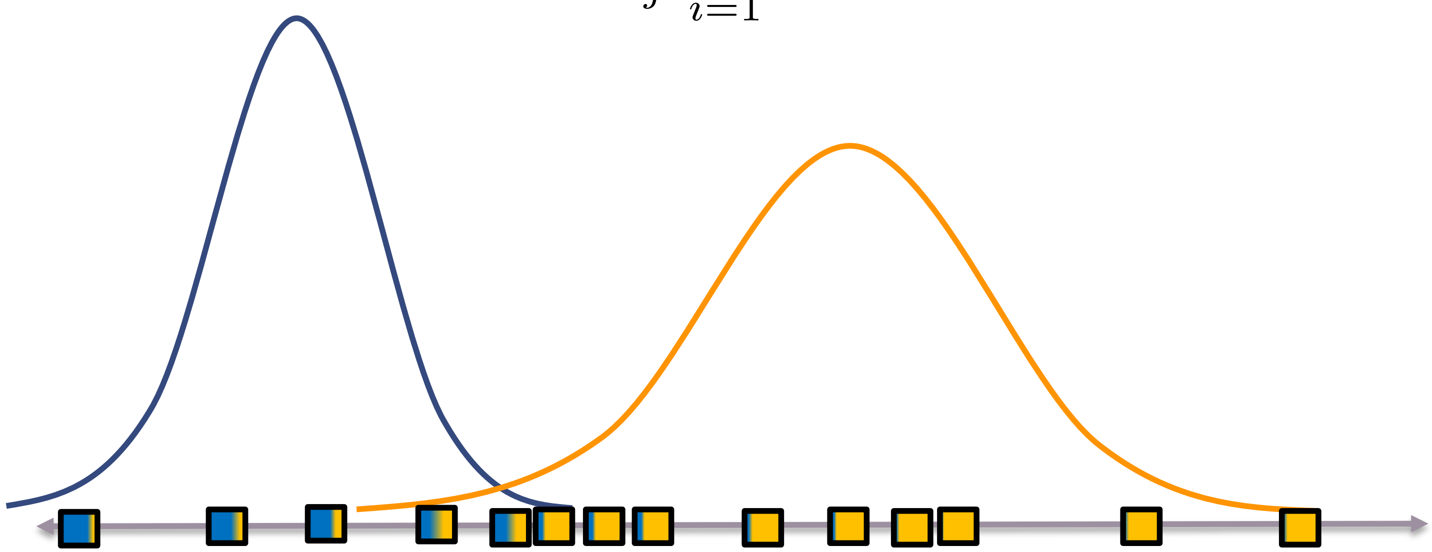
$$P\left(x^{(1)}|\mu^{(1)}, \sigma_1^2\right) = \frac{1}{\sqrt{2\pi\sigma_1^2}} \exp\left[-\frac{\left(x^{(1)} - \mu^{(1)}\right)^2}{2\sigma_1^2}\right]$$

Expectation Maximization for GMMs

- **M-Step**: optimizes each cluster separately given $p(j|i)$

$$\hat{n}_j = \sum_{i=1}^{n} p(j|i) \qquad \hat{\bar{\mu}}^{(j)} = \frac{1}{\hat{n}_j} \sum_{i=1}^{n} p(j|i) \bar{x}^{(i)}$$

$$\hat{\gamma}_j = \frac{\hat{n}_j}{n} \qquad \hat{\sigma}_j^2 = \frac{1}{d\hat{n}_j} \sum_{i=1}^{n} p(j|i) ||\bar{x}^{(i)} - \hat{\bar{\mu}}^{(j)}||^2$$



Expectation Maximization for GMMs: M step (note correspondence with known labels)

if you knew the "soft" cluster assignment $p(j|i)$,

you could compute MLE parameters $\bar{\theta}$ as follows

MLE for GMM with known labels

$\hat{n}_j = \sum_{i=1}^{n} \delta(j \mid i)$

$$\hat{n}_j = \sum_{i=1}^{n} p(j|i)$$

effective number of points assigned to cluster j

$\gamma_j = \frac{\hat{n}_j}{n}$

$$\hat{\gamma}_j = \frac{\hat{n}_j}{n}$$

"fraction" of points assigned to cluster j

$\bar{\mu}^{(j)} = \frac{1}{\hat{n}_j} \sum_{i=1}^{n} \delta(j \mid i) \ \bar{x}^{(i)}$

$$\hat{\bar{\mu}}^{(j)} = \frac{1}{\hat{n}_j} \sum_{i=1}^{n} p(j|i) \bar{x}^{(i)}$$

weighted mean of points in cluster j

$\sigma_j^2 = \frac{1}{d\hat{n}_j} \sum_{i=1}^{n} \delta(j \mid i) \left\| \bar{x}^{(i)} - \bar{\mu}^{(j)} \right\|^2$

$$\hat{\sigma}_j^2 = \frac{1}{d\hat{n}_j} \sum_{i=1}^{n} p(j|i) ||\bar{x}^{(i)} - \hat{\bar{\mu}}^{(j)}||^2$$

weighted spread in cluster j

Expectation Maximization for GMMs
M-step: Example

- **M-Step**: optimizes each cluster separately given $p(j|i)$

$$\hat{n}_j = \sum_{i=1}^{n} p(j|i) \qquad \hat{\gamma}_j = \frac{\hat{n}_j}{n} \qquad \hat{\bar{\mu}}^{(j)} = \frac{1}{\hat{n}_j}\sum_{i=1}^{n} p(j|i)\bar{x}^{(i)} \qquad \hat{\sigma}_j^2 = \frac{1}{d\hat{n}_j}\sum_{i=1}^{n} p(j|i)||\bar{x}^{(i)} - \hat{\bar{\mu}}^{(j)}||^2$$

$\hat{n}_1 =$

$\hat{\gamma}_1 =$

$\hat{\bar{\mu}}^{(1)} =$

Example

Datapoints	Cluster 1
$\bar{x}^{(1)} = [0,1]^T$	0.2
$\bar{x}^{(2)} = [2,1]^T$	0.1
$\bar{x}^{(3)} = [1,1]^T$	0.4
$\bar{x}^{(4)} = [0,2]^T$	0.7
$\bar{x}^{(5)} = [2,2]^T$	0.8

Expectation Maximization for GMMs
M-step: Example

- **M-Step**: optimizes each cluster separately given $p(j|i)$

$$\hat{n}_j = \sum_{i=1}^{n} p(j|i) \quad \hat{\gamma}_j = \frac{\hat{n}_j}{n} \quad \hat{\bar{\mu}}^{(j)} = \frac{1}{\hat{n}_j}\sum_{i=1}^{n} p(j|i)\bar{x}^{(i)} \quad \hat{\sigma}_j^2 = \frac{1}{d\hat{n}_j}\sum_{i=1}^{n} p(j|i)||\bar{x}^{(i)} - \hat{\bar{\mu}}^{(j)}|$$

$$\hat{n}_1 = 0.2 + 0.1 + 0.4 + 0.7 + 0.8 = 2.2$$

$$\hat{\gamma}_1 = \frac{\hat{n}_1}{n} = \frac{2.2}{5} = 0.44$$

Example

Datapoints	Cluster 1
$\bar{x}^{(1)} = [0,1]^T$	0.2
$\bar{x}^{(2)} = [2,1]^T$	0.1
$\bar{x}^{(3)} = [1,1]^T$	0.4
$\bar{x}^{(4)} = [0,2]^T$	0.7
$\bar{x}^{(5)} = [2,2]^T$	0.8

$$\hat{\bar{\mu}}^{(1)} = \frac{1}{\hat{n}_1}\sum_{i=1}^{5} p(1|i)\ \bar{x}^{(i)}$$

$$= \frac{1}{2.2}\Big(p(1|1)\ \bar{x}^{(1)} + p(1|2)\,\bar{x}^{(2)} + p(1|3)\ \bar{x}^{(3)} + p(1|4)\ \bar{x}^{(4)}$$
$$+ p(1|5)\ \bar{x}^{(5)}\Big)$$

$$= \frac{1}{2.2}(0.2[0,1]^T + 0.1[2,1]^T + 0.4[1,1]^T + 0.7[0,2]^T + 0.8[2,2]^T)$$

Similarly compute $\hat{\sigma}_1^2 = \frac{1}{d\hat{n}_1}\sum_{i=1}^{5} p(1|i)\ \left\|\bar{x}^{(i)} - \hat{\bar{\mu}}^{(1)}\right\|^2$

Expectation Maximization for GMMs

EM algorithm for GMM:

initialize parameters

- **E-step**: softly assign points to clusters according to posterior prob

$$p(j|i) = \frac{\gamma_j P(\bar{x}^{(i)}|\bar{\mu}^{(j)}, \sigma_j^2)}{P(\bar{x}^{(i)}|\bar{\theta})}$$

- **M-Step**: optimizes each cluster separately given $p(j|i)$

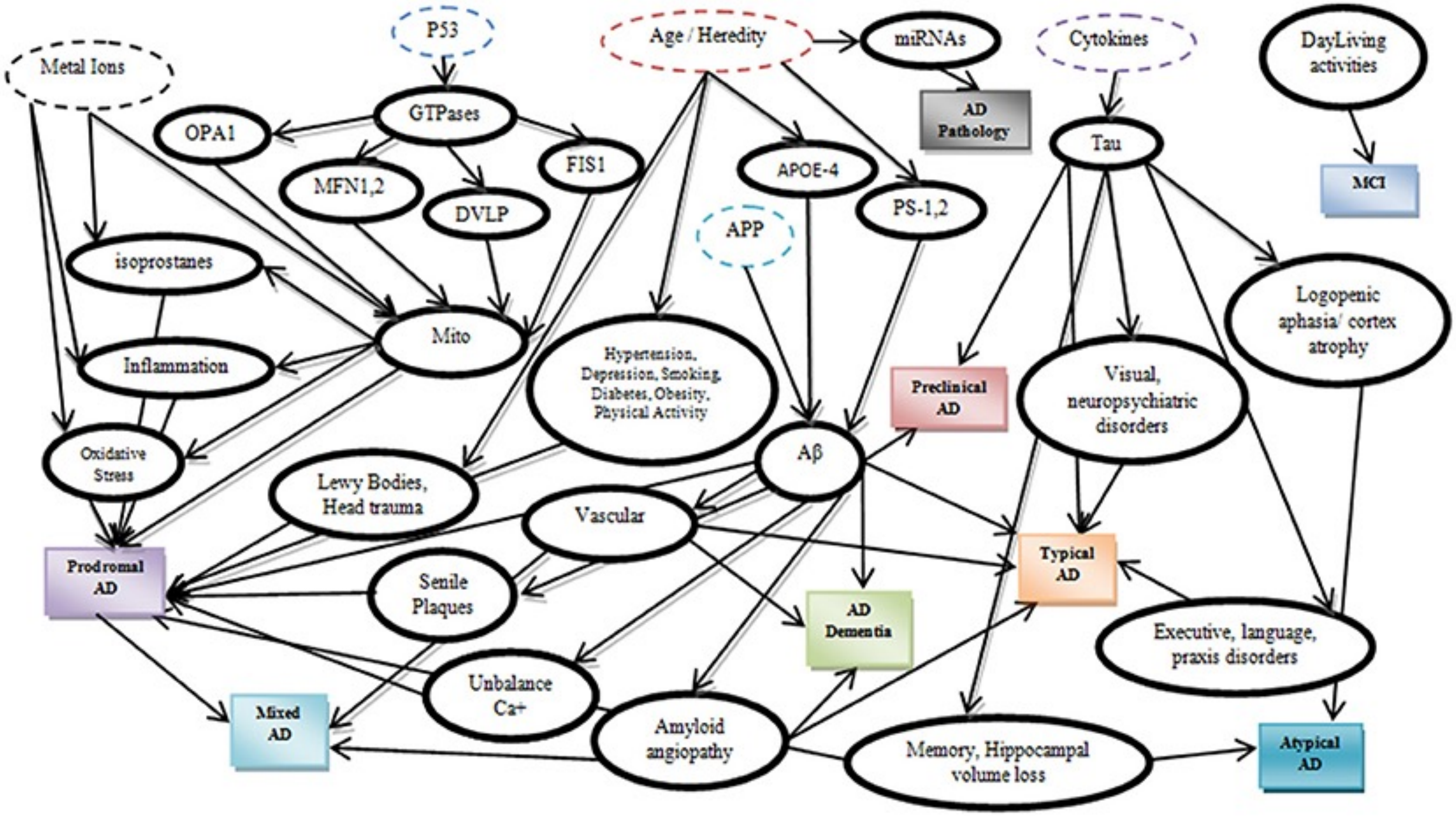
$$\hat{n}_j = \sum_{i=1}^{n} p(j|i) \qquad \hat{\bar{\mu}}^{(j)} = \frac{1}{\hat{n}_j} \sum_{i=1}^{n} p(j|i) \bar{x}^{(i)}$$

$$\hat{\gamma}_j = \frac{\hat{n}_j}{n} \qquad \hat{\sigma}_j^2 = \frac{1}{d\hat{n}_j} \sum_{i=1}^{n} p(j|i) ||\bar{x}^{(i)} - \hat{\bar{\mu}}^{(j)}||^2$$

Iterate until convergence

Graphical Models:
Bayesian Networks

Bayesian Networks: Applications

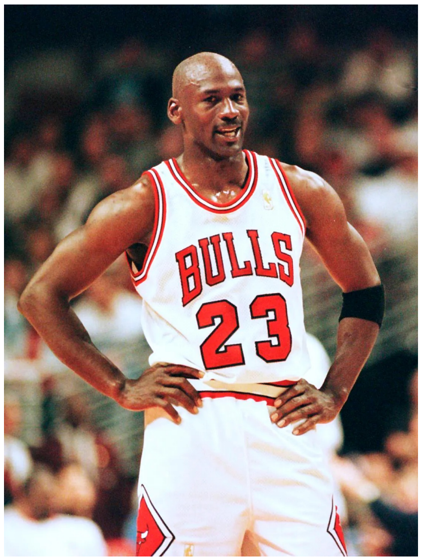


Alexiou Athanasios, Mantzavinos Vasileios D., Greig Nigel H., Kamal Mohammad A. [2017]
A Bayesian Model for the Prediction and Early Diagnosis of Alzheimer's Disease

Bayesian Networks

"Graphical models are a marriage between probability theory and graph theory."

Michael Jordan



not this Michael Jordan

this one

A Bayesian network is a probabilistic graphical model that represents a set of variables and their conditional dependencies via a directed acyclic graph (DAG).

Why use Bayesian Networks?

- savings in number of parameters
- Also captures dependencies which makes it easier to learn and infer
- For inference
  - Can compute marginal probabilities
    - $\Pr(x_3)$
  - Can compute conditional probabilities
    - $\Pr(x_3|x_1)$

Bayesian Networks by Example

1 row

x_1

H	T
0.5	0.5

x_2

H	T
0.5	0.5

1 row

$x_3 \colon x_1 == x_2$

joint probability distribution: $\Pr(x_1, x_2, x_3) = \Pr(x_1)\Pr(x_2|x_1)\Pr(x_3|x_1, x_2)$

Bayesian Networks by Example

nodes: variable

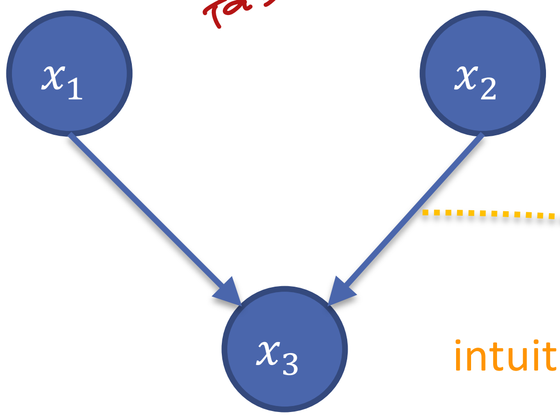
directed edges: dependencies

CPT
Conditional
Probability
Table

1 row 

x_1

H	T
0.5	0.5



x_2

H	T
0.5	0.5

 1 row

intuitively, read this edge as “influences”

x_1 is a parent of x_3

x_3 is a child of x_1

$x_3 : x_1 == x_2$

x_1	x_2	$\Pr(x_3 = T \mid \boldsymbol{x_1}, \boldsymbol{x_2})$	$\Pr(x_3 = \boldsymbol{F} \mid \boldsymbol{x_1}, \boldsymbol{x_2})$
H	H	1	0
T	H	0	1
H	T	0	1
T	T	1	0

Factorization *based on given graph*: $\Pr(x_1, x_2, x_3) = \Pr(x_1)\Pr(x_2)\Pr(x_3 \mid x_1, x_2)$

Bayesian Networks: factorization

- Recall chain rule of probability:

  $$\Pr(X_1, \dots, X_d)$$
  $$= \Pr(X_1 | X_2, \dots, X_d) \Pr(X_2 | X_3 \dots, X_d) \dots \Pr(X_{d-1} | X_d) \Pr(X_d)$$

- Bayesian Networks encode conditional independencies

- Thus, for a given graph, the joint distribution can be written as a *product of the conditional probability of each variable given its parents*.
  - Variables $X_1, \dots, X_d$
  - Parents of variable X_i represented by pa_i

$$P(X_1, ..., X_d) = \prod_{i=1}^{d} P(X_i | X_{pa_i})$$

Two notions of Independence

$Pr(x_1, x_2) = Pr(x_1 | x_2) Pr(x_2)$

Marginal independence

$$\Pr(X_1, X_2) = \Pr(X_1)\Pr(X_2)$$

Conditional independence

$$\Pr(X_1, X_2 | X_3) = \Pr(X_1 | X_3)\Pr(X_2 | X_3)$$

$$\boxed{X_1 \perp X_2 | X_3}$$

Alternately, $\Pr(X_1 | X_2, X_3) = \Pr(X_1 | X_3)$

Bayesian Networks encode independencies

d-separation: Inferring independence

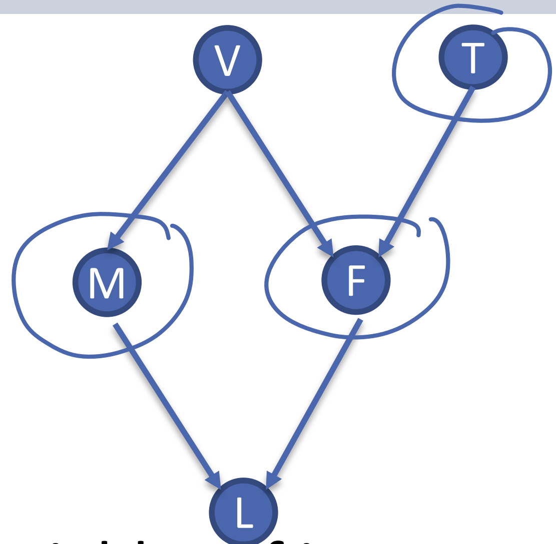
Bayesian Networks provide us a way to determine these via the dependency graph

Inferring independence properties

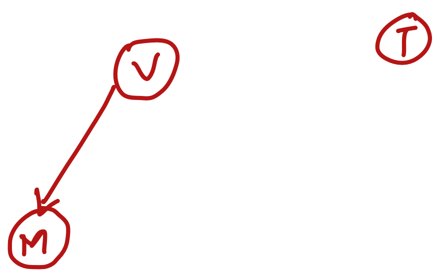
Does d-separation imply:

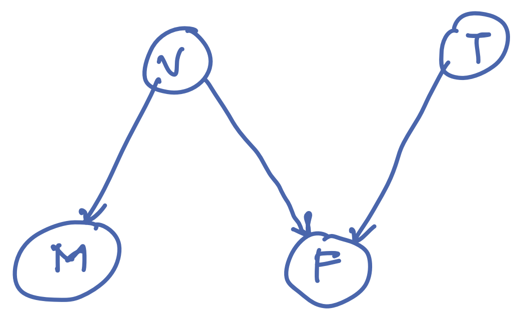
1 $M \perp T?$

2 $M \perp T \mid F?$



Step 1: keep only "ancestral" graph of the variables of interest



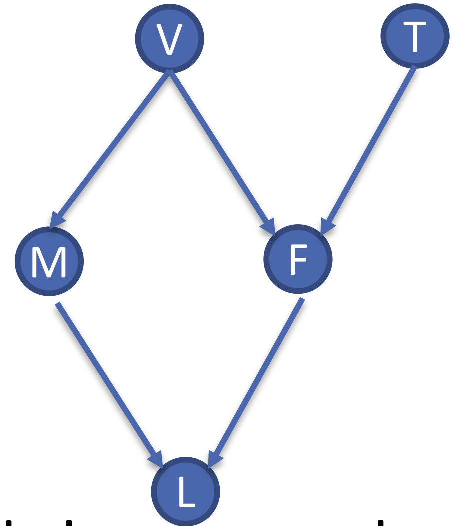


Inferring independence properties

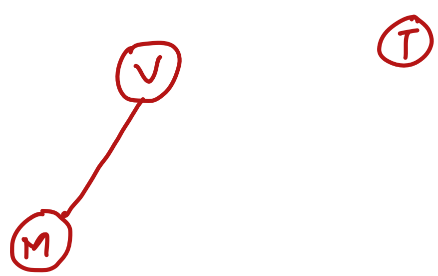
Does d-separation imply:

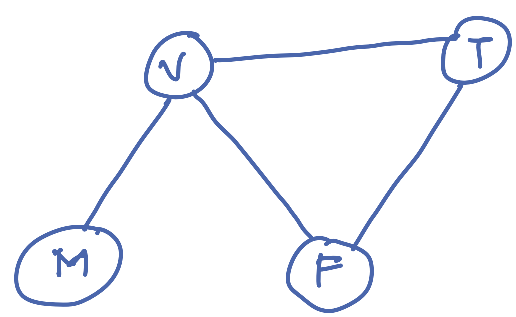
$M \perp T?$

$M \perp T \mid F?$



<u>Step 2</u>: connect nodes with common child and change graph to undirected

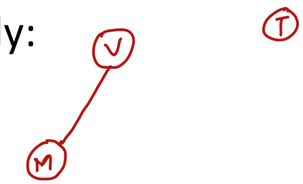




* if multiple parents connect pairwise

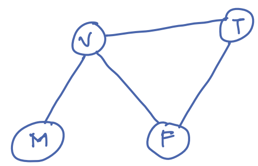
Inferring independence properties

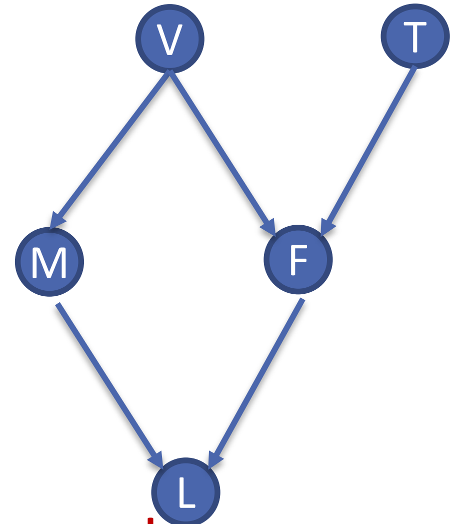
Does d-separation imply:



$M \perp T$? Yes

$M \perp T \mid F$? No





If there is no path between variables of interest, then they are marginally independent

If all paths between variables of interest go through a particular node, then the variables are independent given that node

intuitively can say that that node “blocks” the influence from the first variable to the second