Expectation Maximization

1. 비어있는 것을 채우기 위해 local search
2. K-means… EM algorithm…

Note: this material partly contains the slides provided by Prof. Padhraic Smyth

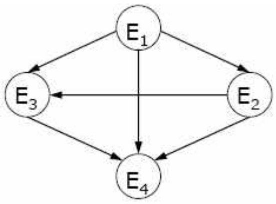
Learning With Hidden Variables

- Why do we want hidden variables?
- Simple case of missing data
- EM algorithm
- Bayesian networks with hidden variables

2

Hidden variables

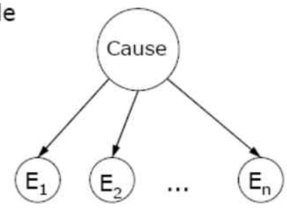
Without the cause, all the evidence is dependent on each other



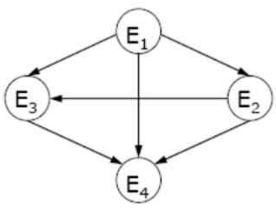
$O(2^n)$ parameters

Hidden variables

Cause is unobservable



$O(n)$ parameters



$O(2^n)$ parameters

Without the cause, all the evidence is dependent on each other

Missing Data

A	B
1	1
1	1
0	0
0	0
0	0
0	H
0	1
1	0

hidden
(unknown)

- Given two variables, no independence relations
- Some data are missing
- Estimate parameters in joint distribution
- Data must be missing at random

여기서 7개는 fully..... 어쩌구
0 H 만 언노운
-> 어떻게 처리해야하는가?

Ignore it

A	B
1	1
1	1
0	0
0	0
0	0
0	H
0	1
1	0

Estimated Parameters

	~A	A
~B	3/7	1/7
B	1/7	2/7

	~A	A
~B	.429	.143
B	.143	.285

Ignore it

A	B
1	1
1	1
0	0
0	0
0	0
0	H
0	1
1	0

Estimated Parameters

	~A	A
~B	3/7	1/7
B	1/7	2/7

	~A	A
~B	.429	.143
B	.143	.285

$$\log \Pr(D|M) = \log(\Pr(D, H=0 \mid M) + \Pr(D, H=1 \mid M))$$
$$= 3\log .429 + 2\log .143 + 2\log .285 + \log(.429 + .143)$$
$$= -9.498$$

이 확률 계산해보면
마이너스값 나온다

Fill in With Best Value

A	B
1	1
1	1
0	0
0	0
0	0
0	0
0	1
1	0

Estimated Parameters

일단 찍어본다
A 0 일때 B 0인게 더 많으니까…

해보구 표 채우면 이렇게된다

	~A	A
~B	4/8	1/8
B	1/8	2/8

	~A	A
~B	.5	.125
B	.125	.25

Fill in With Best Value

A	B
1	1
1	1
0	0
0	0
0	0
0	0
0	1
1	0

Estimated Parameters

	~A	A
~B	4/8	1/8
B	1/8	2/8

	~A	A
~B	.5	.125
B	.125	.25

$$\begin{aligned}\log \Pr(D|M) &= \log(\Pr(D,H=0 \mid M) + \Pr(D,H=1 \mid M)) \\ &= 3\log .5 + 2\log .125 + 2\log .25 + \log(.5 + .125) \\ &= -9.481\end{aligned}$$

아까보다 조금 더 커졌다

Fill in With $\boxed{\text{Distribution}}$

추전해보자!

A	B
1	1
1	1
0	0
0	0
0	0
0	H
0	1
1	0

Guess a distribution over A,B and compute a distribution over H

θ_0

	~A	A
~B	.25	.25
B	.25	.25

$$\begin{aligned}\Pr(H|D,\theta_0) &= \Pr(H \mid D^6, \theta_0) \\ &= \Pr(B \mid \neg A, \theta_0) \\ &= \Pr(\neg A, B \mid \theta_0) / \Pr(\neg A \mid \theta_0) \\ &= .25/0.5 \\ &= 0.5\end{aligned}$$

Fill in With Distribution

A	B
1	1
1	1
0	0
0	0
0	0
0	0, 0.5개 1, 0.5개
0	1
1	0

Use distribution over H to compute better distribution over A,B

Maximum likelihood estimation using *expected counts*

Fill in With Distribution

A	B
1	1
1	1
0	0
0	0
0	0
0	0, 0.5 1, 0.5
0	1
1	0

Use distribution over H to compute better distribution over A,B

Maximum likelihood estimation using *expected counts*

θ_1

	~A	A
~B	3.5/8	1/8
B	1.5/8	2/8

0 1 인 경우
7번째 0 1
6번째 0 1 (0.5개)

	~A	A
~B	.4375	.125
B	.1875	.25

Fill in With Distribution

A	B
1	1
1	1
0	0
0	0
0	0
0	
0	1
1	0

확률 파라미터(테이블)을 이용해서
이게 0 0, 0 1이 될 확률을 다시 추정한다

Use new distribution over AB to get a better distribution over H

θ_1

	~A	A
~B	.4375	.125
B	.1875	.25

$$\Pr(H|D,\theta_1) = \Pr(\neg A, B \mid \theta_1) / \Pr(\neg A \mid \theta_1)$$
$$= .1875 / .625$$
$$= 0.3$$

Fill in With Distribution

A	B
1	1
1	1
0	0
0	0
0	0
0	0, 0.7 1, 0.3
0	1
1	0

바뀌니까~ 파라미터 다시 업데이트

Use distribution over H to compute better distribution over A,B

θ_2

	~A	A
~B	3.7/8	1/8
B	1.3/8	2/8

	~A	A
~B	.4625	.125
B	.1625	.25

Fill in With Distribution

A	B
1	1
1	1
0	0
0	0
0	0
0	
0	1
1	0

Use new distribution over AB to get a better distribution over H

θ_2

	~A	A
~B	.4625	.125
B	.1625	.25

$$\Pr(H|D,\theta_2) = \Pr(\neg A, B \mid \theta_2) / \Pr(\neg A \mid \theta_2)$$
$$= .1625/.625$$
$$= 0.26$$

Fill in With Distribution

A	B
1	1
1	1
0	0
0	0
0	0
0	0, 0.74 1, 0.26
0	1
1	0

Use distribution over H to compute better distribution over A,B

θ_3

	~A	A
~B	3.74/8	1/8
B	1.26/8	2/8

	~A	A
~B	.4675	.125
B	.1575	.25

3페이지정도 해보면
모노톤 increasing 나온다

-> 한 점으로 수렴하게 된다
그러므로 이 방식은 항상 좋아지는 방향으로만 가고 있다
그래서 이 것을 search algorithm에서 local algorithm이라고 말할 수 있다
(좋은 쪽으로 가는거니까)

Increasing Log-Likelihood

θ_0

	~A	A
~B	.25	.25
B	.25	.25

$$\log \Pr(D \mid \theta_0) = -10.3972$$

ignore: -9.498
best val: -9.481

θ_1

	~A	A
~B	.4375	.125
B	.1875	.25

$$\log \Pr(D \mid \theta_1) = -9.4760$$

θ_2

	~A	A
~B	.4625	.125
B	.1625	.25

$$\log \Pr(D \mid \theta_2) = -9.4524$$

θ_3

	~A	A
~B	.4675	.125
B	.1575	.25

$$\log \Pr(D \mid \theta_3) = -9.4514$$

한 세번 돌리니까 많이 비슷해진다

수렴한 자체가 우리가 찾고자 하는 값이다
0 : 0.74
1 : 0.26

EM Algorithm

- Pick initial θ_0
- Loop until apparently converged

EM Algorithm

- Pick initial θ_0
- Loop until apparently converged 뺑뺑이 돌리는 법
  - $\tilde{P}_{t+1}(H) = \Pr(H \mid D, \theta_t)$ 계에속 위아래위아래 반복
  - $\theta_{t+1} = \underset{\theta}{\arg\max}\, E_{\tilde{P}_{t+1}} \log \Pr(D, H \mid \theta)$

EM Algorithm

- Pick initial θ_0
- Loop until apparently converged
  - $\tilde{P}_{t+1}(H) = \Pr(H \mid D, \theta_t)$ Expectation
  - $\theta_{t+1} = \underset{\theta}{\arg\max}\, E_{\tilde{P}_{t+1}} \log \Pr(D, H \mid \theta)$ Maximization
- Monotonically increasing likelihood
- Convergence is hard to determine due to plateaus
- Problems with local optima

EM for Bayesian Networks

- D: observable variables
- H: values of hidden variables in each case
- Assume structure is known
- Goal: maximum likelihood estimation of CPTs

EM for Bayesian Networks

- D: observable variables
- H: values of hidden variables in each case
- Assume structure is known
- Goal: maximum likelihood estimation of CPTs

- Initialize CPTs to anything (with no 0's)
- Fill in the data set with distribution over values for hidden variables
- Estimate CPTs using expected counts

Filling in the data

- Distribution over H factors over the M data cases

$$\tilde{P}_{t+1}(H) = \Pr(H \mid D, \theta_t)$$
$$= \prod_m \Pr(H^m \mid D^m, \theta_t)$$

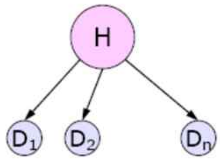
Filling in the data

- Distribution over H factors over the M data cases

$$\tilde{P}_{t+1}(H) = \Pr(H \mid D, \theta_t)$$
$$= \prod_m \Pr(H^m \mid D^m, \theta_t)$$

- We really just need to compute a distribution over each individual hidden variable
- Each factor is a call to Bayes net inference

EM for BN: Simple Case

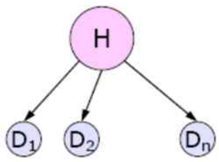


EM for BN: Simple Case

D_1	D_2	...	D_n	$\Pr(H^m \mid D^m, \theta_t)$
1	1		0	.9
0	1		0	.2
0	0		1	.1
1	0		1	.6
1	1		1	.2
1	1		1	.5
0	1		0	.3
0	0		0	.7
1	1		0	.2

Bayes net inference

확률 값을 어떻게 계산하느냐~



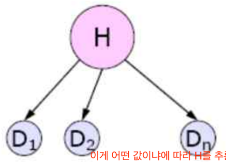
EM for BN: Simple Case

H값을 추론한다

Bayes net inference

D_1	D_2	...	D_n	$\Pr(H^m \mid D^m, \theta_t)$
1	1		0	.9
0	1		0	.2
0	0		1	.1
1	0		1	.6
1	1		1	.2
1	1		1	.5
0	1		0	.3
0	0		0	.7
1	1		0	.2

채워놓는다



이게 어떤 값이냐에 따라 H를 추론하게 된다

$$E\#(H) = \sum_m \Pr(H^m \mid D^m, \theta_t) = 3.7$$

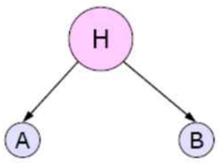
$$E\#(H \wedge D_2) = \sum_m \Pr(H^m \mid D^m, \theta_t) I(D_2^m) = .9 + .2 + .2 + .5 + .3 + .2 = 2.3$$

$$\Pr(D_2 \mid H) \approx 2.3/3.7 = .6216$$

Re-estimate θ

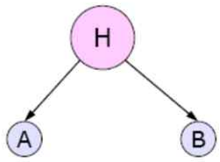
EM for BN: Worked Example

A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	
0	1	1	
1	0	1	
1	1	4	



EM for BN: Worked Example

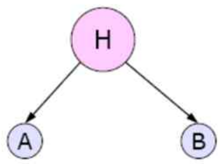
A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	
0	1	1	
1	0	1	
1	1	4	



$$\theta_1 = \Pr(H)$$
$$\theta_2 = \Pr(A \mid H)$$
$$\theta_3 = \Pr(A \mid \neg H)$$
$$\theta_4 = \Pr(B \mid H)$$
$$\theta_5 = \Pr(B \mid \neg H)$$

EM for BN: Initial Model

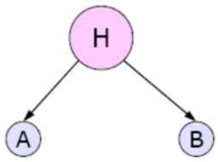
A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	
0	1	1	
1	0	1	
1	1	4	



$$\Pr(H) = 0.4$$
$$\Pr(A|H) = 0.55$$
$$\Pr(A|\neg H) = 0.61$$
$$\Pr(B|H) = 0.43$$
$$\Pr(B|\neg H) = 0.52$$

Iteration 1: Fill in data

A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	.48
0	1	1	.39
1	0	1	.42
1	1	4	.33



$$\Pr(H) = 0.4$$ 다 더한 것

$$\Pr(A|H) = 0.55$$

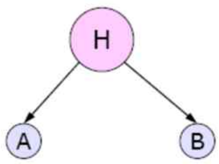
$$\Pr(A|\neg H) = 0.61$$

$$\Pr(B|H) = 0.43$$

$$\Pr(B|\neg H) = 0.52$$

Iteration 1: Re-estimate Params

A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	.48
0	1	1	.39
1	0	1	.42
1	1	4	.33



$$\Pr(H) = 0.42$$

$$\Pr(A|H) = 0.35$$

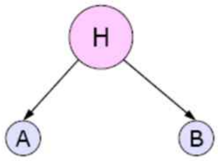
$$\Pr(A|\neg H) = 0.46$$

$$\Pr(B|H) = 0.34$$

$$\Pr(B|\neg H) = 0.47$$

Iteration 2: Fill in Data

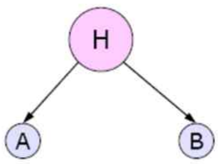
A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	.52
0	1	1	.39
1	0	1	.39
1	1	4	.28



$$\Pr(H) = 0.42$$
$$\Pr(A|H) = 0.35$$
$$\Pr(A|\neg H) = 0.46$$
$$\Pr(B|H) = 0.34$$
$$\Pr(B|\neg H) = 0.47$$

Iteration 2: Re-estimate params

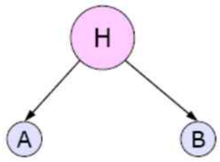
A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	.52
0	1	1	.39
1	0	1	.28
1	1	4	.28



$$\Pr(H) = 0.42$$
$$\Pr(A|H) = 0.31$$
$$\Pr(A|\neg H) = 0.50$$
$$\Pr(B|H) = 0.30$$
$$\Pr(B|\neg H) = 0.50$$

Iteration 5

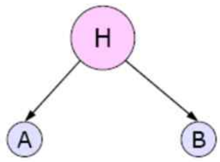
A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	.79
0	1	1	.31
1	0	1	.31
1	1	4	.05



$$\Pr(H) = 0.46$$
$$\Pr(A|H) = 0.09$$
$$\Pr(A|\neg H) = 0.69$$
$$\Pr(B|H) = 0.09$$
$$\Pr(B|\neg H) = 0.69$$

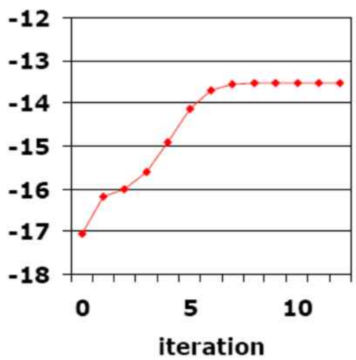
Iteration 10

A	B	#	$\Pr(H^m \mid D^m, \theta_t)$
0	0	6	.971
0	1	1	.183
1	0	1	.183
1	1	4	.001



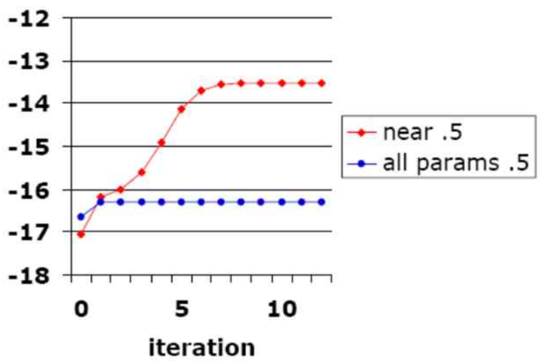
$$\Pr(H) = 0.52$$
$$\Pr(A|H) = 0.03$$
$$\Pr(A|\neg H) = 0.83$$
$$\Pr(B|H) = 0.03$$
$$\Pr(B|\neg H) = 0.83$$

Increasing Log Likelihood



7,8 번째에 수렴이 된다

Increasing Log Likelihood

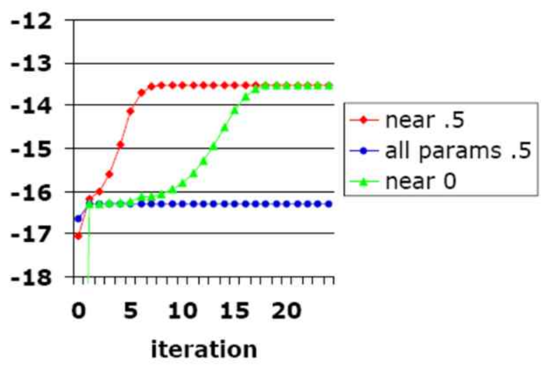


0.5로 해서 표현하면
성능이 낮은데..

어느지점에ㅓ 시작하냐에 따라 수렴하는 지점이 달라진다 (l ocal search의 특징)
::
이걸 해결하는 방법이
랜덤하게 이니셜라이즈해서 여러번 리스타트하는 것

이 파란색도 그 점을 보여준다
무조건 0.5로 하면… 수렴하는 지점이 성능이 좋지 않다는걸 알게 된다
0.5근처에 있게끔 다시 설정해줘야 한다

Increasing Log Likelihood



EM in BN issues

- With multiple hidden nodes, take advantage of conditional independencies
- Lots of tricks to speed up computation of expected counts

EM in BN issues

- With multiple hidden nodes, take advantage of conditional independencies
- Lots of tricks to speed up computation of expected counts 비어있어도 얼마든지 추론해서 써먹을 수 있다
- If structure is unknown, add search operators to add and delete hidden nodes
- There are clever ways of search with unknown structure and hidden nodes
- EM Alogrithm Demo
  - http://the-wabe.com/notebook/em-applet.html