2/4/13

Review

* Learning in BNs
* ML estimation from complete data examples $\{(x_1^{(t)}, x_2^{(t)}, \ldots\ldots, x_n^{(t)}\}_{t=1}^{T}$

t	X_1	X_2	...	X_n
1	1	2	5	3
2	2	?	0	3
3	?	1	1	8
⋮	?	?	5	5
T	2	6	3	9

?- hidden values

$$P_{ML}(x_i = x \mid pa_i = \pi) = \frac{\text{count}(x_i = x, pa_i = \pi)}{\text{count}(pa_i = \pi)}$$

$$= \frac{\sum_{t=1}^{T} I(x^{(t)}, x)\, I(pa_i^{(t)}, \pi)}{\sum_{t=1}^{T} I(pa_i^{(t)}, \pi)}$$

I: indicator ~~functions~~ function test for equality

* ML estimation from incomplete data
  Examples $t = 1, 2, \ldots T$
  Hidden nodes $H^{(t)}$
  Visible nodes $V^{(t)}$
  Choose CPTs to maximize log-likelihood :

$$\left.\begin{aligned} \mathcal{L} &= \sum_{t=1}^{T} \log P(V = v^{(t)}) \\ &= \sum_{t} \log \sum_{h} P(V = v^{(t)}, H = h) \\ &= \sum_{t} \log \sum_{h} \prod_{i=1}^{n} P(x_i = x \mid pa_i = \pi) \Big|_{V = v^{(t)}, H = h} \end{aligned}\right\} \text{How to maximize?}$$

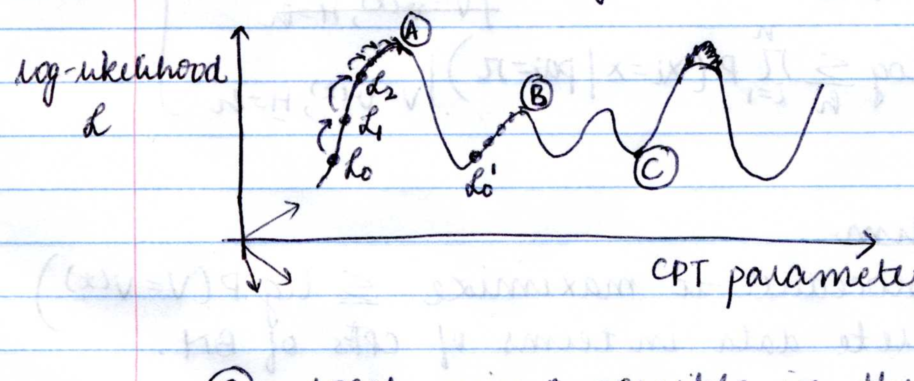
* EM algorithm
  Iterative procedure to maximize $\sum \log P(V = v^{(t)})$ for incomplete data in terms of CPTs of BN.
* Pseudo code
  1) Initialize all CPTs with (possibly) random values.
  2) Do until log-likelihood stops increasing :
     (A) E-step : compute posterior probabilities
     $P(x_i = x, pa_i = \pi \mid V = v^{(t)})$ for $t = 1, \ldots T$
     (B) M-step : update CPTs (non-inference algorithm)
     $$P(x_i = x \mid pa_i = \pi) \leftarrow \frac{\sum_t P(x_i = x, pa_i = \pi \mid V^{(t)})}{\sum_t P(pa_i = \pi \mid V^{(t)})}$$

Intuitions:
- expected statistics under $P(H/V^{(t)})$ are filling in "missing values" of incomplete data.
- expected counts are substituting for observed counts in complete data case.

Iterate E&M steps until convergence. Why iterate?
- RHS depends on current CPTs.

Key properties
- monotonic convergence
  Each iteration of EM improves log-likelihood
  $\mathcal{L} = \sum_t \log P(V^{(t)})$

If $\mathcal{L}_k$ is log-likelihood at k^{th} iteration, then $\mathcal{L}_k \geq \mathcal{L}_{k-1}$.
converges to stationary point of log-likelihood
where gradient vanishes



(A) global maximum: most desirable, not guaranteed

(B) local max: usual outcome

(C) local min: possible in theory, but never in practice.

Why is EM popular?

→ No tuning parameters: no step sizes, learning rates, no backtracking, ...

$\boxed{\text{Example}}$ $A \to B \to C$ A and C are observed (visible)
B is hidden.

* Posterior probability

$$P(B=b \mid A=a, C=c) = \frac{P(C=c \mid B=b, A=a)\, P(B=b \mid A=a)}{\sum_{b'} P(C=c \mid B=b', A=a)\, P(B=b' \mid A=a)}$$ Bayes rule

$$= \frac{P(C=c \mid B=b)\, P(B=b \mid A=a)}{\sum_{b'} P(C=c \mid B=b')\, P(B=b' \mid A=a)}$$ conditional independence

Shorthand: $P(b \mid a, c) = P(B=b \mid A=a, C=c)$

* Incomplete data set $\{(a_t, c_t)\}_{t=1}^{T}$

t	A	B	C
1	a_1	?	c_1
2	a_2	?	c_2
⋮		⋮	⋮
T	a_T	?	c_T

log-likelihood:

$$\mathcal{L} = \sum_t \log P(A=a_t, C=c_t)$$

$$= \sum_t \log P(A=a_t, C=c_t)$$

$$= \sum_t \log \sum_b P(A=a_t, B=b, C=c_t)$$ marginalization

$$= \sum_t \log \left\{ \sum_b P(a_t)\, P(b \mid a_t)\, P(c_t \mid b) \right\}$$ product rule and CI

General EM algorithm:

$$P(x_i = x \mid pa_i = \pi) \longleftarrow \frac{\sum_t P(x_i = x, pa_i = \pi \mid V^{(t)})}{\sum_t P(pa_i = \pi \mid V^{(t)})}$$

Now apply to this example...

M-step : $P(B=b / A=a) \leftarrow \dfrac{\sum_t P(A=a, B=b / A=a_t, C=c_t)}{\sum_t P(A=a / A=a_t, C=c_t)}$

Simplify : $P(b/a) \leftarrow \dfrac{\sum_t I(a,a_t)\, P(b/a_t, c_t)}{\sum_t I(a,a_t)}$

We showed how to compute these from CPTs using Bayes rule.

$$P(C=c/B=b) \leftarrow \frac{\sum_{t=1}^{T} P(B=b, C=c \mid A=a_t, C=c_t)}{\sum_{t=1}^{T} P(B=b \mid A=a_t, C=c_t)}$$

Simplify: $P(c/b) \leftarrow \dfrac{\sum_{t=1}^{T} I(c,c_t)\, P(b/a_t, c_t)}{\sum_{t=1}^{T} P(b/a_t, c_t)}$ computed in terms of CPTs using Bayes rule.

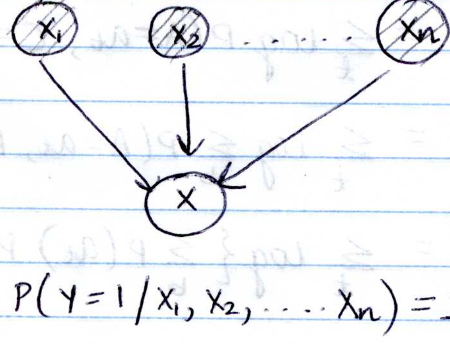
Aside :

$P(B=b, C=c \mid A=a_t, C=c_t)$

$= P(C=c \mid A=a_t, C=c_t)\; P(B=b / A=a, C=c_t, C=c)$

$= I(c, c_t)\; P(B=b / A=a_t, C=c_t)$

$\boxed{\text{Noisy-OR model}}$



disease $X_i \in \{0,1\}$

symptom $Y \in \{0,1\}$

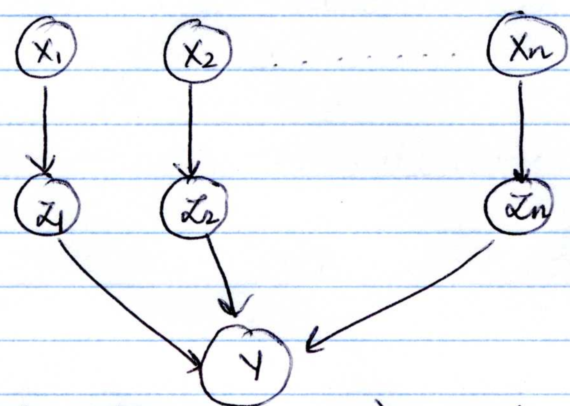
$$P(Y=1 / X_1, X_2, \ldots X_n) = 1 - \prod_{i=1}^{n} (1-p_i)^{X_i} \quad \text{where } p_i \in [0,1]$$

* For complete data $\{(\vec{X}_t, Y_t)\}_{t=1}^{T}$ how to estimate $p_i \in [0,1]$?

Note : NOISY-OR is a "parametric" model of CPT.

No simple, "closed-form" ML estimate for $p_i \in [0,1]$

* Alternative formulation



$P(Y=1 / Z_1, Z_2, \ldots Z_n) = OR(Z_1, Z_2, \ldots Z_n)$ deterministic

$P(Z_i=1 / X_i=0) = 0$ copy X_i if $X_i = 0$

$P(Z_i=1 / X_i=1) = p_i$ flip X_i with prob $1-p$ if $X_i=1$

Are these models equivalent?