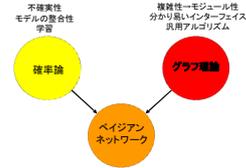


情報意味論 (9) ベイジアンネットワーク

慶應義塾大学工学部
櫻井 彰人

どこから生まれてきたか？

- 実問題の共通課題：
 - 不確実性 ← 確率的枠組み ← 確率変数を用いよう
 - 複雑性 ← すっきりと表現しよう

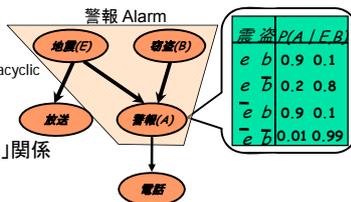


BNとは何か？

条件付確率を用いた、結合確率のコンパクトな表現

定性的要素:

- 有向無閉路グラフ directed acyclic graph (DAG)
- ノード – 確率変数.
- エッジ – 非「条件付独立」関係



あわせて:

ある確率分布の因数分解(?、確率分布の積に分解)

定量的要素:
条件付確率分布の集まり

Figure from N. Friedman

なぜ役立つか？

- グラフ構造があるので
 - 知識をモジュール化して表現できる
 - 推論・学習に、局所的かつ分散的アルゴリズムが使える
 - 直感的な (場合によっては因果的な) 解釈が可能
- 結合確率 $P(X_1, \dots, X_n)$ をそのまま表現するより、指数関数的に少ないパラメータで、表現可能
 - 学習に必要なデータ数 (sample complexity) が少なくてすむ
 - 推論に必要な時間 (time complexity) が少なくてすむ

何に使うか？

■ 事後確率推定

- 証拠・現象 evidence から発生した事象 event の確率を推定

これは、全確率変数の結合確率が分かっている場合

■ 最も可能性が高い説明

- 証拠・現象を説明するシナリオ

これは、因果関係的な解釈ができる場合

■ 合理的な意思決定

- 期待成果を最大化
- 情報の価値

これは、全確率変数の結合確率が分かっている場合



Figure from N. Friedman

応用事例

- “Microsoft’s competitive advantage lies in its expertise in Bayesian networks”
-- Bill Gates, LA Times より, 1996
- MS Answer Wizards, (printer) troubleshooters
- 医療診断
- 遺伝子系統解析
- 音声認識 (HMMs)
- 遺伝子配列分析
- Turbocodes (通信路の符号化)

条件付確率表 CPT

A	false	0.6
A	true	0.4

P(B A)		A	
B	false	0.01	0.7
	true	0.99	0.3

P(C B)		B	
C	false	0.40	0.90
	true	0.60	0.10

各ノード X_i には条件付確率表 $P(X_i | \text{Parents}(X_i))$ があり、親ノードの当該ノードへの影響を表現する

表中の条件付確率がパラメータとなる。

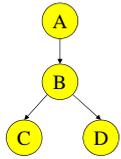
$$P(D, C, B, A) = P(D|B)P(C|B)P(B|A)P(A)$$

P(D B)		B	
D	false	0.02	0.05
	true	0.98	0.95

BNの定義 (まとめると)

BNの構成要素:

1. 有向無閉路グラフ DAG directed acyclic graph



3. 全変数の結合確率は、各ノードに付随する条件付確率の積

$$P(D, C, B, A) = P(D|B)P(C|B)P(B|A)P(A)$$

もし構造がなければ $\Pr(D|A, B, C) \cdot \Pr(C|A, B) \cdot \Pr(B|A) \cdot \Pr(A)$

2. 各ノードに付随する条件付確率表

A	false	0.6
A	true	0.4

P(B A)		A	
B	false	0.01	0.7
	true	0.99	0.3

P(D B)		B	
D	false	0.02	0.05
	true	0.98	0.95

P(C B)		B	
C	false	0.40	0.90
	true	0.60	0.10

条件付確率表 CPT

親ノードがBであるノードC, の条件付確率表

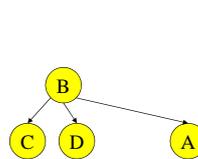
		B	
C	false	0.4	0.9
	true	0.6	0.1

ここには直接の親ノードしか現れない

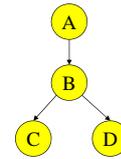
親ノード(左図ではB)の値のすべての組み合わせについて、 $P(C=true|B)$ と $P(C=false|B)$ の和は1とならないといけない

k 個の親が全てブール変数(2値変数)であれば、ブール値変数のCPTの要素数は $2^k * 2 = 2^{k+1}$ となる

補足: naïve Bayes との比較



$$P(D, C, B, A) = P(D|B)P(C|B)P(A|B)P(B) = P(D|B)P(C|B)P(A, B)$$

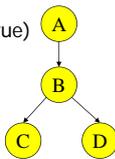


$$P(D, C, B, A) = P(D|B)P(C|B)P(B|A)P(A) = P(D|B)P(C|B)P(A, B)$$

計算例

先ほどの例で次の結合確率を計算する:

$$\begin{aligned} P(A = \text{true}, B = \text{true}, C = \text{true}, D = \text{true}) &= P(A = \text{true}) * P(B = \text{true} | A = \text{true}) * \\ &P(C = \text{true} | B = \text{true}) * P(D = \text{true} | B = \text{true}) \\ &= (0.4) * (0.3) * (0.1) * (0.95) \end{aligned}$$

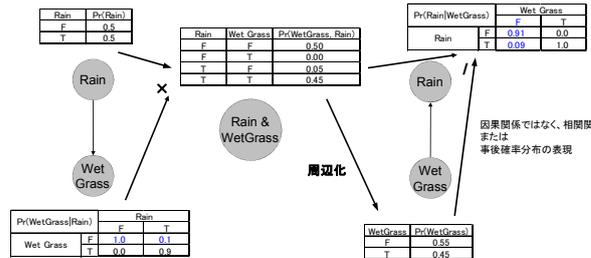


例題が簡単すぎて、あまり簡単にならないが、...

別の計算例

$$\Pr(R = a | WG = b) = \frac{\Pr(R = a, WG = b)}{\Pr(WG = b)}$$

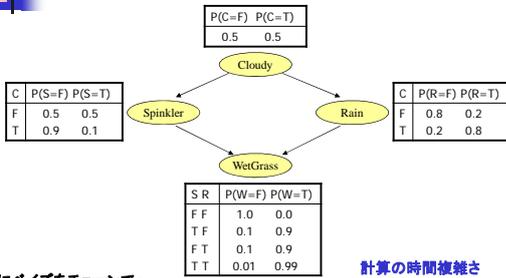
$$\Pr(WG = a, R = b) = \Pr(WG = a | R = b) \cdot \Pr(R = b)$$



因果関係ではなく、相関関係または事後確率分布の表現

$$\Pr(WG = a) = \sum_b \Pr(WG = a, R = b)$$

他の例：Water-Sprinkler



単にベイズをチェーンで：
 $\Pr(C, R, S, W) = \Pr(C) \cdot \Pr(R|C) \cdot \Pr(S|R, C) \cdot \Pr(W|R, C, S)$ $2 \times 4 \times 8 \times 16 = 1024$

条件付独立性を使うと：
 $\Pr(C, R, S, W) = \Pr(C) \cdot \Pr(R|C) \cdot \Pr(S|C) \cdot \Pr(W|R, S)$ $2 \times 4 \times 4 \times 8 = 256$

計算の時間複雑さ

条件付独立性の判定方法

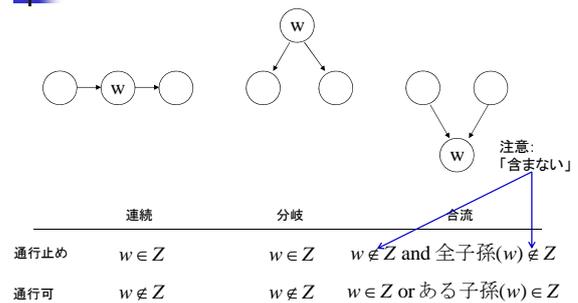
全変数の結合確率表を作って計算すれば分かるのだが、それはたくない

- D-separation: ある証拠が与えられたとき、それに対応する変数を条件として、他の変数が条件付独立であるための十分条件を与える。
 - 証拠: ある確率変数達について、実現した値
- DAG上で、2変数間を、証拠変数がさえぎるか否かを判定し、それで、条件付独立か否かを表している。

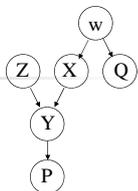
D-separation

- D-separation は、DAG上の変数間の独立性を調べるグラフ的なテストである
- A, B: 変数集合. 独立性を調べる
Z: 変数集合. 条件
A の全ての変数とBの全ての変数間の全てのpathを調べる
- AとBはZを条件として (i.e. Zが観測されるとき) 独立である ($A \perp\!\!\!\perp B \mid Z$) iff A の全ての変数とBの全ての変数の間の全てのpathが通行止めである
- もしpathが一つでも通行可能であれば、独立も非独立もいえない
- D-separationが成立していないときに独立性を言おうと思えば、条件付確率表を調べるしかない
- ある pathが通行止めであるのは、このpath上のあるノード列が次のスライドに示す「通行止め」になっている場合である。

通行止め



例



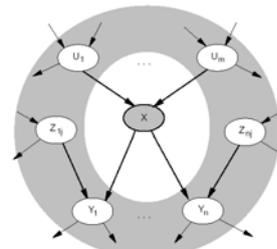
正しい関係

D separation による説明

- $(Q \perp\!\!\!\perp X, Y, Z, P \mid W)$: $Q \leftarrow W \rightarrow X$ は分岐. Wを条件として通行止め
- $(Z \perp\!\!\!\perp X, W, Q \mid \emptyset)$: $Z \rightarrow Y \leftarrow X$ は合流. Y及びその子孫Pを条件としないので通行止め.
- $(Z \perp\!\!\!\perp X, W, Q \mid P)$: $Z \rightarrow Y \leftarrow X$ は合流. Yの子孫Pを条件としているので通行可能.
- $(Z, Y, P \perp\!\!\!\perp W, Q \mid X)$: $W \rightarrow X \rightarrow Y$ は連続. Xを条件として通行止め.
- $(Z, Y, P \perp\!\!\!\perp W, Q \mid \emptyset)$: $W \rightarrow X \rightarrow Y$ は連続. Xを条件としないので通行可能.

Markov Blanket

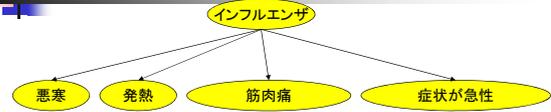
- Markov blanket: 親 + 子供 + 子供の親
- (中心にある)ノードは、Markov blanket 内の変数を条件として、ネットワーク内のどの変数からも、条件付独立である



推論

- ベイジアンネットワークで確率を計算することを推論という
- 一般に、推論では次の形のクエリーが扱われる:
 $P(X | E)$
 - E = 証拠 evidence 変数
 - X = 問い合わせる変数

推論



- クエリーは、例えば、:
 $P(\text{インフルエンザ} = \text{true} | \text{発熱} = \text{true}, \text{急性症状} = \text{true})$
- 注: 悪寒と筋肉痛という変数がベイジアンネットワーク中に現れているが、クエリー中では値が与えられていない (ie. 質問変数としても証拠変数としても現れていない)
- 未観測の確率変数として扱われる

BNにおける推論

他の例: Water-Sprinkler

- WetGrass が真のとき、2つの説明が可能: Rain か Sprinkler

- どちらがよりありうるか?

$$\Pr(S=T | W=T) = \frac{\Pr(S=T, W=T)}{\Pr(W=T)} = \frac{\sum_{C,R} \Pr(C, R, S=T, W=T)}{\Pr(W=T)} = \frac{0.2781}{0.6471} = 0.430 \quad \text{Sprinkler}$$

$$\Pr(R=T | W=T) = \frac{\Pr(R=T, W=T)}{\Pr(W=T)} = \frac{\sum_{C,S} \Pr(C, S, R=T, W=T)}{\Pr(W=T)} = \frac{0.4581}{0.6471} = 0.708 \quad \text{Rain}$$

Rain が真であるのが理由である可能性がより高い

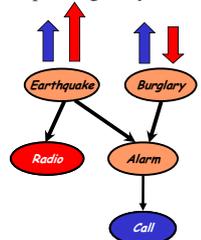
BNにおける推論 (2)

- Bottom-Up :
 - 結果から原因へ → 診断 diagnostic
 - 例. エキスパートシステム, パターン認識...
 - 証拠・結果が与えられたとき、それを説明する最もありうべき仮説を求める
- Top-Down :
 - 原因から結果へ → 推論 causal
 - 例. 生成モデル, 計画...
 - ある仮説のもとどのような結果がどのような確率で起こるか?
- Explain Away :
 - Sprinkler と Rain は, WetGrass が真であることの説明に際し、競合している → この二つは、共通の子供 (WetGrass) が観測されると条件付依存となる

Explaining away effect

ある仮定(または仮定の集合)を支持する証拠が、その証拠とは相容れない(競合する)仮定の確からしさを減少させる効果、またはその現象

Explaining away effect



Call=true が観測されると、Earthquake=true への信頼度も Burglary=true への信頼度も上昇する。しかし、Radio=true がさらに観測されると、Earthquake=true への信頼度は上昇するが、Burglary=true への信頼度は減少する。

推論 – まとめると

- 因果推論 Causal Inferences $E \rightarrow Q \rightarrow O$
- 診断推論 Diagnostic Inferences $Q \rightarrow O \rightarrow E$
- 原因間推論 Intercausal Inferences $Q \rightarrow O$ and $E \rightarrow O$
- 混合推論 Mixed Inferences $E \rightarrow Q \rightarrow E$

推論 – 結局のところ

- 条件付確率を求めること

$$P(Q|E) = \frac{P(Q,E)}{P(E)}$$

Q と E は確率変数(または当該確率変数のある値)の集合で、重なりはない

- そのため、結合確率が高速に計算できるとよい

Naïve な推論

BN で $P(Q|E=e)$ を解く naive なアルゴリズム

- 条件付確率を全て乗じ、全変数に関する結合確率分布を求める

$$P(Q|E) = \frac{P(Q,E)}{P(E)} = \frac{P(Q,E)}{\sum_q P(Q=q,E)}$$

- BN 構造が使用されず、変数が多いときこのアルゴリズムは実効的ではない
- 一般にこの推論は NP-hard

全然、BN ではない。

手計算でやってみよう

因果推論 Causal Inferences

原因から結果への推論

例: 窃盗が入ったとして、

$P(J=\text{true}|B=\text{true})?$

$P(A=t|B=t)$

$$\begin{aligned} &= P(A=t, E=t|B=t) + P(A=t, E=f|B=t) \\ &= P(A=t|E=t, B=t)P(E=t|B=t) + P(A=t|E=f, B=t)P(E=f|B=t) \\ &= (0.95)(0.002) + (0.94)(0.998) \\ &= 0.94 \end{aligned}$$

$P(J|B) = P(J, A|B) + P(J, \neg A|B)$

$$= P(J|A, B)P(A|B) + P(J|\neg A, B)P(\neg A|B)$$

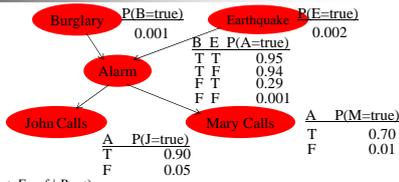
$$= P(J|A)P(A|B) + P(J|\neg A)P(\neg A|B)$$

$$= (0.9)(0.94) + (0.05)(0.06)$$

$$= 0.85$$

略記: A とは $A=t$, $\neg A$ とは $A=f$

同様に $P(M|B)=0.67$ となる



手計算でやってみよう

診断推論 Diagnostic Inferences

結果から原因へ。

例: John が電話をした。では $P(\text{burglary})?$

$P(B|J) = \frac{P(J|B)P(B)}{P(J)}$

$P(J)$ は? まず $P(A)$ が必要:

$$\begin{aligned} P(A) &= P(A, B, E) + P(A, \neg B, E) + P(A, B, \neg E) + P(A, \neg B, \neg E) \\ &= \\ &= 0.002517 \end{aligned}$$

$$P(J) = P(J, A) + P(J, \neg A)$$

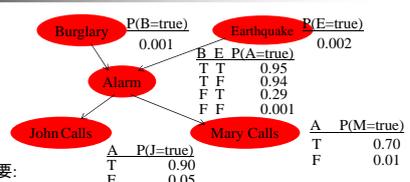
$$=$$

$$=$$

$$= 0.052$$

$$P(B|J) = \frac{(0.85)(0.001)}{(0.052)} = 0.016$$

false positives 多し



手計算でやってみよう

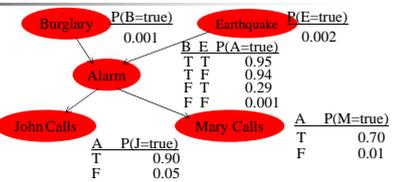
原因間推論 Intercausal Inferences

Explaining away effect が発生する。

Alarm が所与なら, $P(B|A)=0.37$.

そこに Earthquake が真という証拠を加えれば, $P(B|A, E)=0.003$.

すなわち, B と E は独立であるが, A を条件とした条件付独立ではないため、一方に証拠があれば、他方の確率分布は変化する可能性がある



$$P(B, A) =$$

$$=$$

$$= 0.00094002$$

$$P(B|A) = P(B, A)/P(A) = 0.3735$$

$$P(B, E, A) = P(B)P(E)(0.95) = 0.00000019$$

$$P(E, A) =$$

$$=$$

$$= 0.00058132$$

$$P(B|A, E) = P(B, A, E)/P(A, E) = 0.003268$$

手計算でやってみよう

混合推論 Mixed Inferences

原因間推論と診断推論を同時に

例: "John calls" かつ "Earthquake=false":

$P(A|J, \neg E) = 0.03$

$P(B|J, \neg E) = 0.017$

この計算はかなり込み入っている

$P(A, J, \neg E) =$

$=$

$= 0.001742$

$P(\neg A, J, \neg E) =$

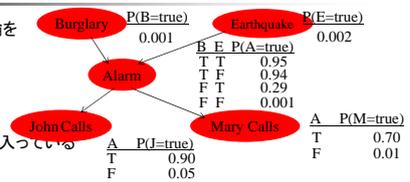
$=$

$=$

$= 0.04980$

$$P(A|J, \neg E) = P(A, J, \neg E) / (P(A, J, \neg E) + P(\neg A, J, \neg E))$$

$$= 0.03379$$



一般化: 行すべき推論

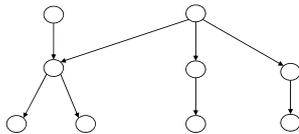
V_{e_1}, \dots, V_{e_n}

- 一部の 변수について、その値が観測される
- 仮に証拠変数と呼ぶ E
- 推論 – 証拠変数以外の 변수 X_i すべてについて、条件付確率 $P(X_i | E)$ を求める
- 一般には、計算量大 – (NP-hard)
- (ある条件のもと) 厳密値の計算方法がある
 - 確率伝播 belief propagation
- 従って、近似計算も用いられる

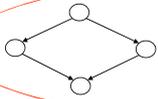
厳密な計算方法 – 信念伝播

- Jude Pearl, 1982 による
 - 単結合グラフ singly-connected graph – どのノード間にもたかだか一つの無向路しか存在しない – についてのアルゴリズム.
 - (下方に、上方に) (確率に基づくある量を送る。これをメッセージと呼ぶ。(原理的には) 収束するまで繰り返す(単結合なら必ず収束する)
 - π -message: ノード X の上方にある証拠(事前分布)による量。下方に送られる
 - λ -message: ノード X の下方にある証拠(事前分布)による量。上方に送られる
- 以下では少し異なる定式化を行う

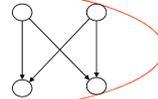
単結合グラフ (または Polytrees)



複数個の親や複数個の子を持つことは可能



条件を満たさず

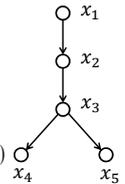


変数の(積分)消去

例: 周辺分布 $p(x_5)$ の計算

$$\begin{aligned}
 p(x_5) &= \sum_{x_1} \sum_{x_2} \sum_{x_3} \sum_{x_4} p(x_1, x_2, x_3, x_4, x_5) \\
 &= \sum_{x_1} \sum_{x_2} \sum_{x_3} p(x_1) p(x_2 | x_1) p(x_3 | x_2) p(x_4 | x_3) p(x_5 | x_3) \\
 &= \sum_{x_1} p(x_1) \sum_{x_2} p(x_2 | x_1) \sum_{x_3} p(x_3 | x_2) \sum_{x_4} p(x_4 | x_3) p(x_5 | x_3)
 \end{aligned}$$

Diagram illustrating the elimination order: x_1, x_2, x_4, x_3, x_5 . The marginalization process is shown with colored bars under the summation indices: $m_{43}(x_3)$ (yellow), $m_{12}(x_2)$ (green), $m_{23}(x_3)$ (cyan), and $m_{35}(x_5)$ (magenta).



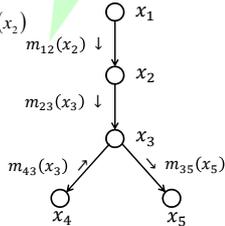
変数消去の順序は: 1, 2, 4, 3

メッセージ伝播

$m_{ij}(x_j)$: i から j へのメッセージと呼ぶ $m_{12}(x_2) = \sum_{x_1} p(x_2 | x_1) p(x_1)$

i は総和をとって消去する変数, j はそれ以外

$$\begin{aligned}
 p(x_5) &= \sum_{x_1} p(x_1) \sum_{x_2} p(x_2 | x_1) \sum_{x_3} p(x_3 | x_2) m_{12}(x_2) \\
 &= \sum_{x_1} p(x_1) \sum_{x_2} p(x_2 | x_1) m_{23}(x_3) \\
 &= \sum_{x_1} p(x_1) m_{23}(x_3) \sum_{x_4} p(x_4 | x_3) \\
 &= \sum_{x_1} p(x_1) m_{23}(x_3) m_{43}(x_3) \\
 &= m_{35}(x_5)
 \end{aligned}$$



消去順序に依存することに注意

信念伝播 (Pearl, 1982)

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \psi_{ij}(x_i, x_j) \prod_{x_k \in N(i) \setminus j} m_{ki}(x_i)$$

i: メッセージ発信元

j: メッセージ送信先

$N(i)$: i の近傍

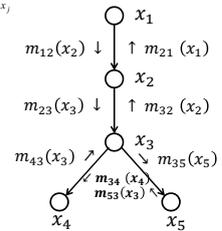
$N(i) \setminus j$: j を除く、i の近傍

例

$$\sum_{x_1} p(x_1) \sum_{x_2} p(x_2 | x_1) m_{23}(x_3) m_{43}(x_3) m_{35}(x_5)$$

周辺分布は:

$$p(x_i) \propto \prod_{x_k \in N(i)} m_{ki}(x_i)$$



但し、Pearl 1982 とは定式化が少し異なる

信念伝播 (Pearl, 1982)

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \psi_{ij}(x_i, x_j) \prod_{x_k \in N(x_i) \setminus x_j} m_{ki}(x_i)$$

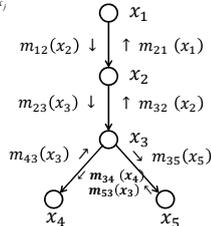
i: メッセージ発信元

j: メッセージ送信先

- (無向な木とした) 葉 i から開始 (葉 = エッジが一つのノード)

$$N(x_i) \setminus \{x_j\} = \phi$$

- 木構造から、各ノード i は、メッセージを j に送る前にすべての $N(x_i) \setminus \{x_j\}$ からメッセージを集めることができる



確率伝播 (和積) 一般化

和積 (sum-product) 更新式

$$m_{ij}(x_j) \leftarrow \alpha \sum_{x_i} \psi_{ij}(x_i, x_j) m_{ki}(x_i) \prod_{x_k \in N(x_i) \setminus x_j} m_{ki}(x_i)$$

$$b_i(x_i) \leftarrow \alpha m_{ii}(x_i) \prod_{x_k \in N(x_i)} m_{ki}(x_i)$$

ただし、 α は正規化定数を表し $N(x_i) \setminus x_j$ は x_i の x_j を除く近傍を表す

$m_{ii}(x_i) = m_{ii}(x_i, y_i)$ は、非観測変数 x_i から観測変数 y_i へのメッセージを表す

確率伝播 (最大-積)

最大-積 (max-product) 更新式

$$m_{ij}(x_j) \leftarrow \alpha \max_{x_i} \psi_{ij}(x_i, x_j) m_{ki}(x_i) \prod_{x_k \in N(x_i) \setminus x_j} m_{ki}(x_i)$$

$$b_i(x_i) \leftarrow \alpha m_{ii}(x_i) \prod_{x_k \in N(x_i)} m_{ki}(x_i)$$

ただし、 α は正規化定数を表し $N(x_i) \setminus x_j$ は x_i の x_j を除く近傍を表す

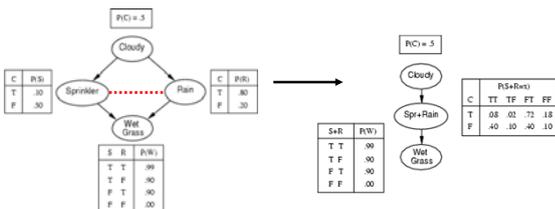
$m_{ii}(x_i) = m_{ii}(x_i, y_i)$ は、非観測変数 x_i から観測変数 y_i へのメッセージを表す

複雑度

- 単結合グラフ (polytree) 上では、BP アルゴリズムは収束する。収束速度はグラフの直径に比例する - 高々線形
- 各ノードごとの作業は CPT のサイズに比例する
- 従って BP の計算量はベイジアンネットワーク中のパラメータ数に対し線形である
- 一般のベイジアンネットワークについては
 - 厳密な推論は NP-hard
 - 近似推論も (まともな近似は) NP-hard

より一般のグラフでは

- 信念伝播法が正しい値に収束するには、グラフが単結合でなければならない
- 一般的なグラフに対しては、それを junction tree に変換してから適用する方法が考えられている
- ただし、計算複雑度は、変換の結果発生するクランプ数の指数オーダーである → もし最適な junction tree を見出そうとすると、それは NP-hard



近似アルゴリズム

- なぜ?
 - ループを含むグラフに対して正確な計算を行おうとすると、指数関数時間かかるため
 - また、連続分布を考えた場合、非ガウスであると、message は閉じた形式では表現できないため
- どうやって?
 - 決定的な近似: loopy BP, 平均場近似 (変分ベイズ) 等
 - 統計的近似: MCMC (ギブスサンプラー), 等

- アルゴリズムにより、速度・精度のトレードオフがある (当然!)

ランダムサンプリング Random Sampling

- For $i = 1$ to n
 1. X_i の親ノード ($X_{p(i,1)}, \dots, X_{p(i,n)}$) を見つける
 2. 当該親ノードにランダムに(このアルゴリズムで)与えられた変数値を読み出す
 3. 次の値を表から読み出す
 $P(X_i | X_{p(i,1)} = x_{p(i,1)}, \dots, X_{p(i,n)} = x_{p(i,n)})$
 4. この確率に従い x_i の値をランダムに設定する

確率的シミュレーション Stochastic Simulation

- 知りたいのは $P(Q = q | E = e)$
- ランダムサンプリングを大量に行い次の個数を数える
 - N_e : $E = e$ となるサンプル数
 - N_q : $Q = q$ かつ $E = e$ となるサンプル数
 - N : ランダムサンプルの総数
- N が充分大きければ
 - N_e / N は $P(E = e)$ の良い推定値
 - N_q / N は $P(Q = q, E = e)$ の良い推定値
 - N_q / N_e は従って $P(Q = q | E = e)$ の良い推定値

補足: 連続変数値

- 条件付確率表を考える場合は、離散変数を仮定している
- 連続値変数に対しては、例えば、ガウス分布を仮定する。その場合、平均値と分散を用いることになる
- しかし、基本的には、離散変数を用いる。実際問題として、連続値であっても離散化することが多いからである。とはいえ、離散化のよしあしが結果に大きく影響するので、簡単ではない。

BNの学習(構築)

- 入出力:
 - 入力: 訓練データと事前知識
 - 出力: ベイジアンネットワーク
 - グラフとパラメータ
- 事前知識:
 - 最善(期待できない): ネットワーク構造
 - 変数間の依存関係
 - 事前分布

場合分け

	構造は既知	構造が未知
完全データ	パラメータの統計的推測 (方程式)	構造を含めて離散最適化 (探索)
不完全データ	パラメータ最適化 (EM, 最急降下,...)	両方 (かなり大変,...)

構築

BN を構築する手続き:

- 適用領域を記述する変数集合を選ぶ
- 変数の順序を定める
- 空のネットワークから開始し、変数をネットワークに、指定した順序に従い、一個ずつ付加していく
- $i=1$ から順に下記を行う
 - 第 i 番目の変数 X_i の付加:
 - すでにネットワーク中にある変数 (X_1, \dots, X_{i-1}) の中の変数から $pa(X_i)$ を
 $P(X_i | X_1, \dots, X_{i-1}) = P(X_i | pa(X_i))$
 となるように定める
 - 領域知識を用いる
 - データから判断する
 - 有向弧を、 $pa(X_i)$ 中の各変数から X_i に結ぶ



例：領域知識を用いて

- 順序: B, E, A, J, M
 - $pa(B)=pa(E)=\{\}$,
 - $pa(A)=\{B,E\}$,
 - $pa(J)=\{A\}$,
 - $pa(M)=\{A\}$
- 順序: M, J, A, B, E
 - $pa(M)=\{\}$,
 - $pa(J)=\{M\}$,
 - $pa(A)=\{M,J\}$,
 - $pa(B)=\{A\}$,
 - $pa(E)=\{A,B\}$
- 順序: M, J, E, B, A
 - 完全に結合したグラフ

例：説明

- 順序: M, J, A, B, E

$P(J|M)$. 簡略化できず
 $P(A|M,J)$. 簡略化できず
 $P(B|M,J,A)$
 $= P(M,J,A,B) / P(M,J,A)$
 $= P(J|A)P(M|A)P(A|B)P(B) / (P(M|A)P(J|A)P(A))$
 $= P(A,B) / P(A)$
 $= P(B|A)$

$P(E|M,J,A,B)$
 $= P(E,M,J,A,B) / P(M,J,A,B)$
 $= P(J|A)P(M|A)P(A|B,E)P(B)P(E) / (P(J|A)P(M|A)P(A|B)P(B))$
 $= P(A,B,E) / P(A,B)$
 $= P(E|A,B)$

変数順序が大切！

どの変数順序を用いるか？

- 視点: 確率を計算する自然な順序. M, J, E, B, A はよくない. なぜなら $P(B | J, M, E)$ は自然でないから
- 視点: 弧の個数の最小化. M, J, E, B, A は宜しくない (弧が多すぎる), 初めの方がよい
- 視点: 因果関係反映, i.e. 原因が結果の前になる. M, J, E, B, A は宜しくない. というのも M と J は A の結果なのに A の前に来ている

領域知識がないとき

- データから判断する.
 - $P(X_i | X_1, \dots, X_{i-1}) = P(X_i | pa(X_i))$ となる最小の $pa(X_i)$ を見つける
 - しかし、データの偏りのため、厳密に上記等号が成立することは期待できない
 - そこで、ある程度のエラーを許容することになる。
 - しかし、どれだけ許容したらよいか分からない。
- 様々な情報量規準を用いる
 - データだけ (多項分布を仮定する (後述) ので、実は頻度) を見ても、データ数の不足・統計的偏りのため、条件付独立性は結論できない。
 - 誤差を見込むことになる。どの程度の誤差なら、「条件付独立」と見なすかという問に対して、それによって、簡単になるなら「条件付独立」と見なそうと答える。
 - その時の、残余誤差と簡単さとの trade-off を考え、判断するために、情報量規準を用いる。
 - MDL やベイズアンネットにおけるその精密化である BD (Bayesian Dirichlet) score がよく用いられる
- 説明は「補足」に

パラメータ学習

- 例:
 - ある BN の構造が所与
 - データ集合

X_1	X_2	X_3	X_4	X_5
0	0	1	1	0
1	0	0	1	0
0	?	0	0	?
...

? は欠測値を表す

- 条件付確率 $P(X_i | pa(X_i))$ の推定

パラメータの推定

- データには欠測値がないとする
- n 変数 X_1, \dots, X_n
- X_i の状態数 or 変数値の数: $r_i = |\Omega_{X_i}|$
- X_i の親変数の状態総数: $q_i = |\Omega_{pa(X_i)}|$
- 推定すべきパラメータ:

$$\theta_{ijk} = P(X_i = j | pa(X_i) = k),$$

$$i = 1, \dots, n; j = 1, \dots, r_i; k = 1, \dots, q_i$$

簡単な例

例: BN を一つ. どの変数も2値 1, 2 をとるとする.

$$\theta_{ijk} = P(X_i = j \mid \text{pa}(X_i) = k)$$

親変数の状態組合せ

$$\theta_{111} = P(X_1=1), \theta_{211} = P(X_1=2)$$

$$\theta_{211} = P(X_2=1), \theta_{221} = P(X_2=2)$$

$$\text{pa}(X_3) = 1: \theta_{311} = P(X_3=1 \mid X_1=1, X_2=1), \theta_{321} = P(X_3=2 \mid X_1=1, X_2=1)$$

$$\text{pa}(X_3) = 2: \theta_{312} = P(X_3=1 \mid X_1=1, X_2=2), \theta_{322} = P(X_3=2 \mid X_1=1, X_2=2)$$

$$\text{pa}(X_3) = 3: \theta_{313} = P(X_3=1 \mid X_1=2, X_2=1), \theta_{323} = P(X_3=2 \mid X_1=2, X_2=1)$$

$$\text{pa}(X_3) = 4: \theta_{314} = P(X_3=1 \mid X_1=2, X_2=2), \theta_{324} = P(X_3=2 \mid X_1=2, X_2=2)$$

要は: 簡単な例

例: BN を一つ. どの変数も2値 1, 2 をとるとする.

$$\theta_{ijk} = P(X_i = j \mid \text{pa}(X_i) = k)$$

親変数の状態組合せ

P(X3 X1,X2)		X1, X2			
		1,1	1,2	2,1	2,2
X3	1	θ_{311}	θ_{312}	θ_{313}	θ_{314}
	2	θ_{321}	θ_{322}	θ_{323}	θ_{324}

P(X3 X1,X2)		X1, X2			
		1,1	1,2	2,1	2,2
X3	1	3	5	7	9
	2	7	15	23	31
		10	20	30	40

サンプル数

P(X3 X1,X2)		X1, X2			
		1,1	1,2	2,1	2,2
X3	1	3/10	5/20	7/30	9/40
	2	7/10	15/20	23/30	31/40

最尤推定

BN におけるパラメータ推定

- 次が求まる:

$$\theta_{ijk}^* = \frac{m_{ijk}}{\sum_j m_{ijk}}$$

- 言葉でいえば,
 $\theta_{ijk} = P(X_i = j \mid \text{pa}(X_i) = k)$ の最尤推定量は

$$\frac{X_i=j \text{ かつ } \text{pa}(X_i) = k \text{ となる事例数}}{\text{pa}(X_i) = k \text{ となる事例数}}$$

しかし、ご存じの通り、ちょっとした問題がある。

BN におけるパラメータ推定

- 実は次の形がよく使われている (Laplace correction):

$$\theta_{ijk}^* = \frac{m_{ijk} + 1}{\sum_j m_{ijk} + r_i}$$

- 言葉でいえば,
 $\theta_{ijk} = P(X_i = j \mid \text{pa}(X_i) = k)$ の最尤推定量は

$$\frac{X_i=j \text{ かつ } \text{pa}(X_i) = k \text{ となる事例数} + 1}{\text{pa}(X_i) = k \text{ となる事例数} + \lceil X_i \text{ の変数値の個数} \rceil}$$

なお、“+1” や “ r_i ” にはもっと一般的な形がある。
 Dirichlet 分布を事前分布とすることに相当する。

補足

ベイジアンネットワークの学習

領域知識がない場合

BNの学習

BNをデータから構成する方法に2種類ある:

- 制約を発見していく方法
 - 統計的検定を行って、条件付独立な変数組を発見していく
 - これを満たす DAG を見つける
- スコア関数を用いる方法
 - DAG を比較するスコア関数を用いる。
eg. Bayesian, BIC, MDL, MML
 - データに最もよくfitする DAG を選ぶ

注: 通常、Markov等価性(説明してありません)による制約を考える。というのも、Markov等価なDAGは統計的には区別できないからである。

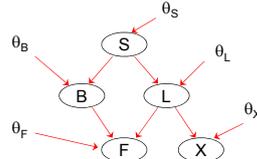
Bayes的方法(1)

(Cooper and Herskovits, 1992)

データを用いて、条件付独立性に関する統計的推定を行う
 - 確率的関係をよりよく表現するモデルを探す

M - 構造を表す離散確率変数。値 m はありうる DAG 構造。
 M の値は分布するとする。確率分布を P(m) で表す。

Θ_m - モデル m に対応した連続ベクトル値の確率変数 (パラメータ)。値 θ_m はそのパラメータ値。 Θ_m の値も分布する。確率分布を P($\theta_m | m$) で表す。



G.F. Cooper and E. Herskovits (1992)
 Machine Learning, 9, 309-47

Bayes的方法(2)

訓練データ集合を D, DAG 構造 m の事後確率は, D が与えられたとして:

$$P(m | D) = \frac{P(m)P(D | m)}{\sum_{m'} P(m')P(D | m')}$$

但し

$$P(D | m) = \int P(D | \theta_m, m)P(\theta_m | m)d\theta_m$$

は周辺尤度である。例によって事前分布 P(m) が一様分布であれば

$$P(m | D) \propto P(D | m)$$

従って、尤度最大化は事後確率最大化となる。

Bayes的方法 (3)

Cooper and Herskovits (1992) によれば、周辺尤度は次の通り

$$P(D | m) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

n - 全ノード数

q_i - ノード X_i の親ノード達の値全部の組合せ総数

r_i - ノード (離散確率変数) X_i の値の総数

α - 事前分布である Dirichlet 分布のパラメータ (i はノード, $1 \leq j \leq q_i$)

N - データ数。ノード i , 親ノード値の組合せ j , k 番目の値

この P(D | m) は Bayesian scoring function として知られている。

G.F. Cooper and E. Herskovits (1992)
 Machine Learning, 9, 309-47

計算例

次の DAG m_1 と訓練データ D を考える



データID	X	Y
1	1	1
2	1	2
3	1	1
4	2	2
5	1	1
6	2	1
7	1	1
8	2	2

P(D | m_1) は

$$P(D | m_1) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(\alpha_{ij} + N_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(\alpha_{ijk} + N_{ijk})}{\Gamma(\alpha_{ijk})}$$

Y ($i=2$) に対し $q_2 = 2$ (X は2値) かつ $r_2 = 2$ (Y は2値). $j = 1$ に対応する項は

$$\frac{\Gamma(2)}{\Gamma(2+5)} \frac{\Gamma(1+4)}{\Gamma(1)} \frac{\Gamma(1+1)}{\Gamma(1)}$$

他の項も計算すれば P(D | m_1) = 7.22×10^{-6}

R.E. Neapolitan, Learning Bayesian Networks (2004)

計算例 (続)

m_1 は、変数 X と Y の間に (条件付) 独立性がないことを示す DAG (の Markov 同値クラス) の代表と考えることができる。

m_2 を エッジがない DAG とすると P(D | m_2) = 6.75×10^{-6}



さらに m_1 と m_2 の事前確率は等しい、すなわち P(m_1) = P(m_2) = 0.5 とすると m_1 の事後確率は m_2 の事後確率より大きくなる。

Bayes の定理により

$$\begin{aligned} P(m_1 | D) &= \frac{P(D | m_1)P(m_1)}{P(D | m_1)P(m_1) + P(D | m_2)P(m_2)} \\ &= \frac{7.215 \times 0.5}{7.215 \times 0.5 + 6.7465 \times 0.5} \\ &= \frac{7.215}{13.9615} \approx 0.517 \end{aligned}$$

探索アルゴリズムの必要性

理想的には全 DAG の空間を網羅的に探索し、前述の Bayesian scoring function を最大化する DAG を見つけたい。

しかし、ノード数を大きく (ほんの少し大きく) しただけで、DAG の数は莫大なものとなる:

ノード数	DAG 総数
1	1
2	3
3	25
4	543
5	29,281
10	4.2×10^{18}

様々な発見の方法が開発されている

K2 Algorithm (1)

(Cooper and Herskovits, 1992)

n 変数 $\{X_1, X_2, \dots, X_n\}$ 間に順序があると仮定する。すなわち, $j > i$ ならば, X_i は X_j の親にはなれないとする。

X_2 について

X_2 に親がないとして Bayesian score を求める

X_2 の親が X_1 として Bayesian score を求める。これがより大きければ X_1 から X_2 へのエッジをつける。

X_1 について

X_1 に親がないとして Bayesian score を求める

X_1 に親が一つだけとして Bayesian score を求める。親がない場合より大きい score があればその最大値を与える X_j からのエッジをつける。

次に第二番目の親を選んで同様のことを試みる。これを score が大きくならないまで続ける。

K2 Algorithm (2)

変数の順序を $\{X, Y, Z\}$ とする

