

MADI

Inférences probabilistes dans les réseaux bayésiens

Pierre-Henri WUILLEMIN

DESIR
LIP6
pierre-henri.wuillemin@lip6.fr

Une base de données

Soit une base de données présentée sous la forme d'un fichier tabulaire comportant 4 colonnes.

A	B	C	D
true	false	false	true
true	false	true	true
false	true	false	false
true	true	false	true
true	false	false	false
...

- Il y a répétition d'événements donc fréquences calculables donc représentable par un **modèle probabiliste**
- Chaque événement est identifié par la liste des valeurs des variables A à D : **modèle probabiliste factorisé**
- Peut-on représenter ce système par un réseau bayésien ?



Vers un réseau bayésien (1) : χ^2

Pour **construire** un réseau bayésien (différent de **apprendre**), il faut isoler les indépendances conditionnelles dans ce modèle probabiliste factorisé : le χ^2 !

Soit X et Y deux v.a. binaires,

si $X \perp\!\!\!\perp Y$ alors $\forall i, j, p(X = i, Y = j) = p(X = i) \cdot p(Y = j)$

Dans le cadre d'un test expérimental, on ne peut avoir que des estimations fréquentistes des probabilités : si $X \perp\!\!\!\perp Y$ alors $\forall i, j, p(X = i, Y = j) = \frac{n_{ij}}{n} = p(X = i) \cdot p(Y = j) = \frac{n_{i.}}{n} \cdot \frac{n_{.j}}{n}$

Tester l'indépendance de X et Y revient donc à comparer $\frac{n_{ij}}{n}$ et $\frac{n_{i.}}{n} \cdot \frac{n_{.j}}{n}$.

➡ Définition (χ^2 d'écart à l'indépendance)

$$d^2 = \sum_{i=1}^r \sum_{j=1}^s \frac{\left(n_{ij} - \frac{n_{i.} \cdot n_{.j}}{n}\right)^2}{\frac{n_{i.} \cdot n_{.j}}{n}}$$

alors $d^2 \leq n \cdot \min(s-1, r-1)$ suit une loi du χ^2 .



Vers un réseau bayésien (2) : tableau de contingence

Première étape donc : calculer les n_{ij} : le **tableau de contingence**.
En l'occurrence, pour notre problème, ce sont des n_{ijkl} qu'il faut calculer (4 variables).

En supposant une base de données de 1000 expériences, on trouve :

		A=True		A=False	
		B=True	B=False	B=True	B=False
C=True	D=True	7	77	2	58
	D=False	5	307	2	230
C=False	D=True	65	19	22	14
	D=False	43	77	14	58

On peut vérifier que $7 + 77 + 2 + 58 + 5 + 307 + 2 + 230 + 65 + 19 + 22 + 14 + 43 + 77 + 14 + 58 = 1\,000$



Vers un BN (3) : $A \perp\!\!\!\perp B$?

		\bar{a}		a	
		\bar{b}	b	\bar{b}	b
\bar{c}	\bar{d}	7	77	2	58
	d	5	307	2	230
c	\bar{d}	65	19	22	14
	d	43	77	14	58



Vers un BN (4) : $A \perp\!\!\!\perp C \mid B$?

		\bar{a}		a	
		\bar{b}	b	\bar{b}	b
\bar{c}	\bar{d}	7	77	2	58
	d	5	307	2	230
c	\bar{d}	65	19	22	14
	d	43	77	14	58



Vers un BN (5) : liste d'indépendances

● $A \perp\!\!\!\perp C | B$

● $A \perp\!\!\!\perp D | B$

● $C \perp\!\!\!\perp D | B$



Un réseau bayésien

		\bar{a}		a	
		b	\bar{b}	b	\bar{b}
c	d	7	77	2	58
	\bar{d}	5	307	2	230
\bar{c}	d	65	19	22	14
	\bar{d}	43	77	14	58



Calcul dans un réseau bayésien

- Les quelques manipulations de base :

Marginalisation $\sum_y P(x, y | z) = p(x | z)$

Somme totale $\sum_y P(y | z) = 1$

Décomposition $P(x, y | z) = P(x | y, z) \cdot P(y | z)$

Chain rule $P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | X_1, \dots, X_{i-1})$

Indépendance $X \perp\!\!\!\perp Y | Z \Rightarrow P(x | y, z) = P(x | z)$

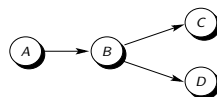
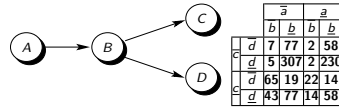
Loi de Bayes $P(x | y, z) \propto P(y | x, z) \cdot P(x | z)$

- Dans un réseau bayésien :

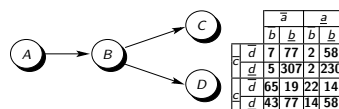
Markov local $P(X_1, \dots, X_n) = \prod_{i=1}^n P(X_i | \text{parents}(X_i))$

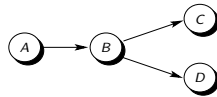


Calcul dans un réseau bayésien (1) : $P(D)$?

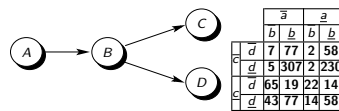


Calcul dans un réseau bayésien (2) : $P(D \mid \underline{a})$?



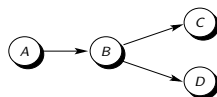


Calcul dans un réseau bayésien (3) : $P(C|\bar{d})$?

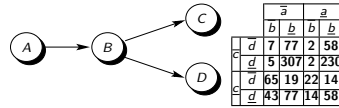


		\bar{a}		a	
		b	\bar{b}	b	\bar{b}
c	\bar{d}	7	77	2	58
	d	5	307	2	230
d	\bar{d}	65	19	22	14
	d	43	77	14	58

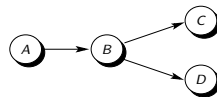
$$P(c | \bar{d})$$



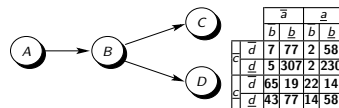
Calcul dans un réseau bayésien (4) : $P(A|\bar{d})$?

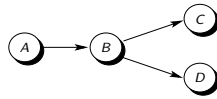


$$P(a | \bar{d})$$



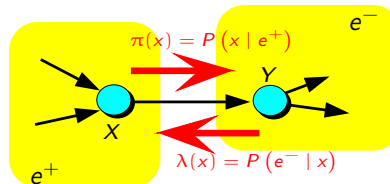
Calcul dans un réseau bayésien (5) : $P(B | \bar{c}, \underline{a})$?



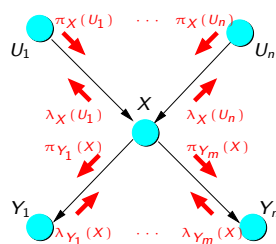


Inférence dans les poly-arbres (graphes orientés sans cycle)

Les messages transitent de nœuds en nœuds dans le sens (π -messages) ou en remontant (λ -messages) les arcs du graphe.



Une propagation de l'ensemble des messages sur la structure du graphe permet à tous les nœuds de connaître l'état global (toute l'information).

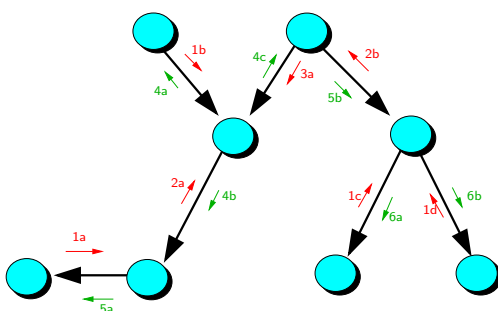


Si un nœud à n parents, il doit connaître les messages issus de $n - 1$ de ses voisins pour pouvoir envoyer le message vers son $n^{\text{ème}}$ voisin.

Algorithme de propagation de messages (version simplifiée sans information)

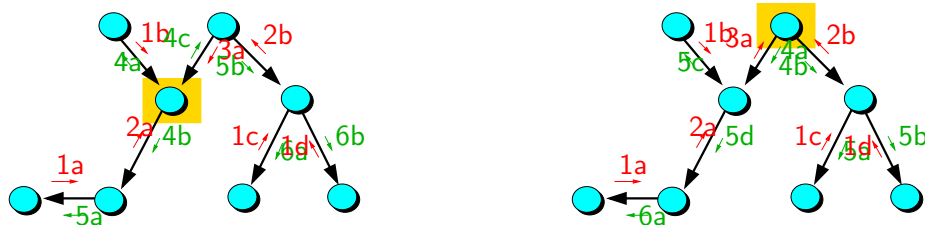
Version asynchrone

- Initialisation
 - les nœuds sans parents peuvent envoyer leurs messages π .
 - les nœuds sans enfants peuvent envoyer leurs messages λ .
- Propagation : pour chaque nœud (avec n voisins),
 - si $n - 1$ messages reçus alors envoi du message vers le $n^{\text{ème}}$ voisin.
 - si n messages reçus alors envoi des messages restants à envoyer vers tous les voisins (et calcul de sa loi a posteriori).



On note que la complexité de cet algorithme est proportionnelle au nombre d'arcs dans le graphe.

Algorithme de propagation de messages (2)



Version centralisée

- Choix d'une racine

Absorption

Tout nœud envoie son message vers la racine dès qu'il le peut.

- Intégration

La racine a reçu les messages de tous ses voisins. Elle envoie donc ses messages vers tous ses voisins.

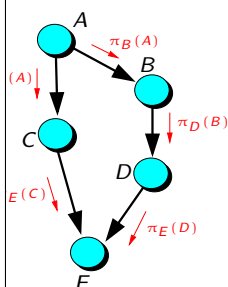
Diffusion

Tout nœud ayant reçu le message venant de la racine a reçu tous ses messages et, donc, peut envoyer tous ses autres messages.



Problèmes dans un graphe avec cycles (mais toujours sans circuit)

$$P(E) = \sum_{A,B,C,D} P(A) \cdot P(B | A) \cdot P(C | A) \cdot P(D | B) \cdot P(E | C, D)$$



En utilisant l'algorithme des messages :

- 1 $\pi_B(A) = P(A)$
- 2 $\pi_C(A) = P(A)$
- 3 $\pi_D(B) = \sum_A P(B | A) \cdot \pi_B(A)$
- 4 $\pi_E(D) = \sum_B P(D | B) \cdot \pi_D(B)$
- 5 $\pi_E(C) = \sum_A P(C | A) \cdot \pi_C(A)$
- 6 $P(E) = \sum_{D,C} P(E | C, D) \cdot \pi_E(C) \cdot \pi_E(D)$

$$P(E) \neq \sum_{*} P(E | C, D) \cdot P(C | A) \cdot P(A) \cdot P(D | B) \cdot P(B | A') \cdot P(A')$$



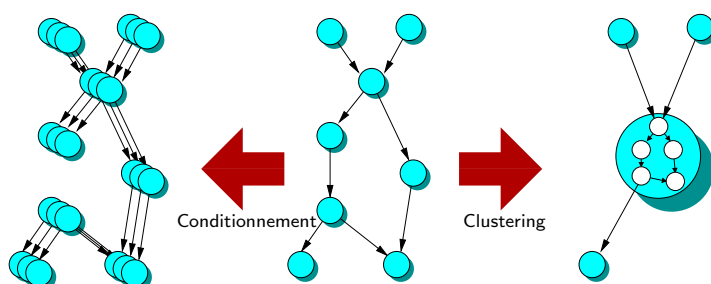
Propager dans des graphes avec cycles (mais toujours sans circuit) ?

La propagation par messages qui permet ne fonctionne que dans un graphe sans cycles.

Se ramener à un graphe sans cycle

Méthodes multiples, par exemple (pour les plus connues) :

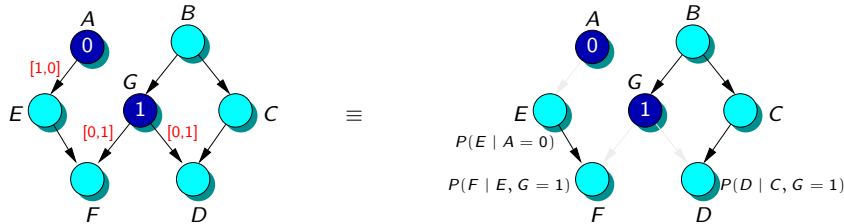
- **Conditioning** : Couper le graphe (retirer des arcs) jusqu'à obtenir un graphe sans cycle.
- **Clustering** : Regrouper (fusionner) les nœuds jusqu'à obtenir un graphe sans cycle.



Conditioning : méthode coupe-cycle

Conditionnement par S

Soit un réseau bayésien sur l'ensemble de variables V . Soient $S \subset V$ et s une instantiation des variables de S . Lors de la propagation de l'information s , **tous les messages π issus d'une variable de S seront déterministes**. Cette propagation est alors équivalente à la propagation dans un réseau bayésien où les arcs issus des nœuds de S seraient supprimés.



Ensemble de coupe

On appelle ensemble de coupe, un ensemble S de nœuds qui permet de supprimer les cycles du réseau bayésien.



Conditioning : méthode coupe-cycle (2)

On rappelle que $\forall S \subset V$, on a la propriété :

$$\forall x \in V, P(x) = \sum_s P(x | s) \cdot P(s)$$

Algorithme du coupe-cycle (global)

Soit un réseau bayésien G ,

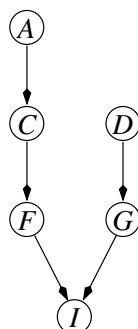
- 1 Soit $S = \{S_1, \dots, S_n\}$ un ensemble de coupe sur G
- 2 $\forall s$ instantiation de S ,
 - Calculer $p_s = P(s)$.
 - Calculer $P_s(x) = P(x | s)$ dans le réseau sans cycle.
- 3 $P(x) = \sum_s (p_s \cdot P_s(x))$

Si on note $\#_i$ le nombre de modalité de la variable S_i , cet algorithme consiste donc à calculer $\prod_i \#_i$ inférences dans un graphe sans cycle.



Osons le calcul des probabilités a priori

$$P(A, C, D, F, G, I) = P(A)P(C|A)P(F|C)P(D)P(G|D)P(I|F, G)$$



Calcul de $P(I)$?



Shafer-Shenoy brut

$$P(A, C, D, F, G, I) = P(A)P(C|A)P(F|C)P(D)P(G|D)P(I|F, G)$$

$$P(I) = \sum_G \left(\sum_F \left(\sum_D \left(\sum_C \left(\sum_A P(A, C, D, F, G, I) \right) \right) \right) \right)$$

$$\sum_A P(A, C, D, F, G, I) = \underbrace{\left(\sum_A P(A)P(C|A) \right)}_{P(C)} P(F|C)P(D)P(G|D)P(I|F, G)$$

$$\sum_C \sum_A P(A, C, D, F, G, I) = \underbrace{\left(\sum_C P(C)P(F|C) \right)}_{P(F)} P(D)P(G|D)P(I|F, G)$$



MADI

Inférences probabilistes dans les réseaux bayésiens

28 / 42

Dissection du produit de deux probabilités

$$P(A, B|C) = \underbrace{\begin{pmatrix} a_1 & a_2 \\ c_1 & c_2 \end{pmatrix}}_{\begin{pmatrix} 0,15 & 0,18 \\ 0,15 & 0,12 \end{pmatrix}} \underbrace{\begin{pmatrix} c_1 & c_2 \\ a_1 & a_2 \end{pmatrix}}_{\begin{pmatrix} 0,07 & 0,56 \\ 0,63 & 0,14 \end{pmatrix}} b_1 = \underbrace{\begin{pmatrix} a_1 & a_2 \\ c_1 & c_2 \end{pmatrix}}_{\begin{pmatrix} 0,5 & 0,6 \\ 0,5 & 0,4 \end{pmatrix}} \underbrace{\begin{pmatrix} c_1 & c_2 \\ a_1 & a_2 \end{pmatrix}}_{\begin{pmatrix} 0,1 & 0,8 \\ 0,9 & 0,2 \end{pmatrix}} b_1 \times \underbrace{\begin{pmatrix} a_1 & a_2 \\ c_1 & c_2 \end{pmatrix}}_{\begin{pmatrix} 0,3 & 0,7 \end{pmatrix}} P(A)$$

$P(B|A, C)$

$$P(I, C|B) = \underbrace{\begin{pmatrix} b_1 & b_2 \\ c_1 & c_2 \end{pmatrix}}_{\begin{pmatrix} 0,48 & 0,08 \\ 0,12 & 0,32 \end{pmatrix}} \underbrace{\begin{pmatrix} c_1 & c_2 \\ b_1 & b_2 \end{pmatrix}}_{\begin{pmatrix} 0,48 & 0,08 \\ 0,12 & 0,32 \end{pmatrix}} i_1 = \underbrace{\begin{pmatrix} a_1 & a_2 \\ c_1 & c_2 \end{pmatrix}}_{\begin{pmatrix} 0,8 & 0,2 \\ 0,2 & 0,8 \end{pmatrix}} i_1 \times \underbrace{\begin{pmatrix} c_1 & c_2 \\ a_1 & a_2 \end{pmatrix}}_{\begin{pmatrix} 0,6 & 0,4 \end{pmatrix}} P(C)$$



MADI

Inférences probabilistes dans les réseaux bayésiens

29 / 42

Shafer-Shenoy graphique (1/6)

Séquence d'élimination

A C D F G

$$P(A, C, D, F, G, I) = P(A)P(C|A)P(F|C)P(D)P(G|D)P(I|F, G)$$

$$\begin{array}{cccccc} A & AC & FC & D & GD & IFG \\ P(A) & P(C|A) & P(F|C) & P(D) & P(G|D) & P(I|F, G) \end{array}$$

$$\text{somme sur } A \Rightarrow P(C) = \sum_A P(A)P(C|A)$$



MADI

Inférences probabilistes dans les réseaux bayésiens

30 / 42

Shafer-Shenoy graphique (2/6)

Séquence d'élimination

A **C** D F G

$$P(C, D, F, G, I) = P(C)P(F|C) P(D)P(G|D)P(I|F, G)$$

$$P(A)P(C|A)$$



FC

D

GD

IFG

P(C)

P(F|C)

P(D)

P(G|D)

P(I|F, G)

$$\text{somme sur } C \Rightarrow P(F) = \sum_C P(C)P(F|C)$$



Shafer-Shenoy graphique (3/6)

Séquence d'élimination

A C **D** F G

$$P(D, F, G, I) = P(F) P(D)P(G|D) P(I|F, G)$$

$$P(A)P(C|A)$$



$$P(F|C)$$



D

GD

IFG

P(F)

P(D)

P(G|D)

P(I|F, G)

$$\text{somme sur } D \Rightarrow P(G) = \sum_D P(D)P(G|D)$$



Shafer-Shenoy graphique (4/6)

Séquence d'élimination

A C D **F** G

$$P(F, G, I) = P(F)P(I|F, G) P(G)$$

$$P(A)P(C|A)$$



$$P(F|C)$$



D

GD

IFG

P(F)

P(D)

P(G|D)

P(I|F, G)

$$\text{somme sur } F \Rightarrow P(I|G) = \sum_F P(F)P(I|F, G)$$

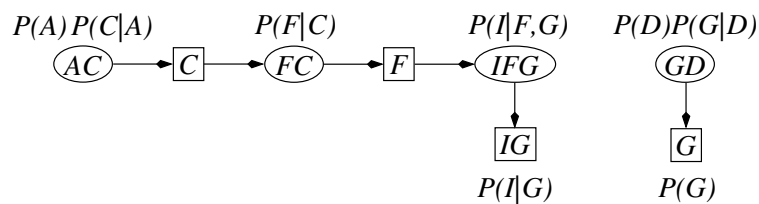


Shafer-Shenoy graphique (5/6)

Séquence d'élimination

A C D F **G**

$$P(G, I) = P(I|G)P(G)$$

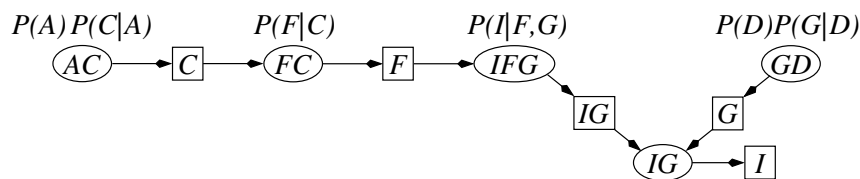


$$\text{somme sur } G \Rightarrow P(I) = \sum_G P(G)P(I|G)$$



Shafer-Shenoy graphique (6/6)

Le graphe final obtenu par Shafer-Shenoy



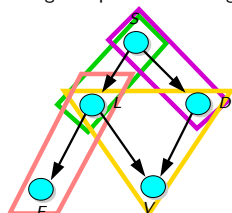
Algorithme de Shafer-Shenoy

- 1 Se donner une séquence d'élimination des nœuds
 \Rightarrow join tree,
- 2 propager les impacts dans le sens des flèches :
 - dans les ellipses (cliques), on effectue des multiplications,
 - dans les rectangles (séparateurs), on effectue des additions (projections).

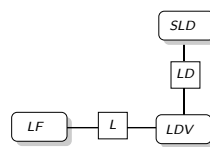
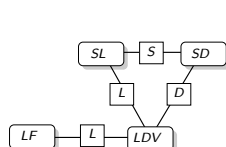


Clustering : arbre de jonction et algorithme de Jensen

Il s'agit d'opérer sur des regroupements de variables... Mais lesquels ?



$$\begin{aligned}
 P(s, l, d, f, v) &= P(s) \cdot P(l | s) \cdot P(d | s) \cdot P(f | l) \cdot P(v | d, l) \\
 &= f(s, l) \cdot g(s, d) \cdot h(l, f) \cdot k(d, l, v) \\
 &= j(s, l, d) \cdot g(s, d) \cdot h(l, f) \cdot k(d, l, v)
 \end{aligned}$$



Arbre de jonction



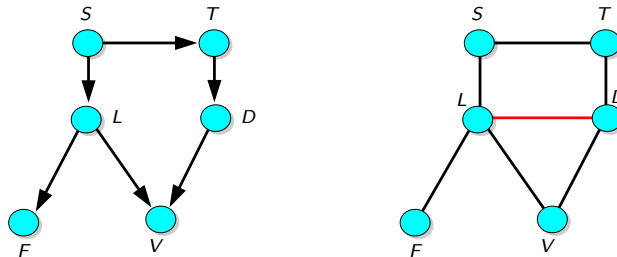
Construction de l'arbre de jonction

Idée : création d'un graphe non-orienté à partir du réseau bayésien, dont les **cliques** (ensembles complets maximaux du graphe) seront les nœuds de l'arbre de jonction.

Les termes $P(X | \text{Parent}_X)$ de la factorisation récursive font apparaître des sous-cliques nécessaires.

Moralisation

Le graphe moral d'un réseau bayésien est le graphe non orienté sous-jacent au réseau bayésien, auquel on ajoute des arêtes liant les parents de chaque nœud.



MADI

Inférences probabilistes dans les réseaux bayésiens

37 / 42

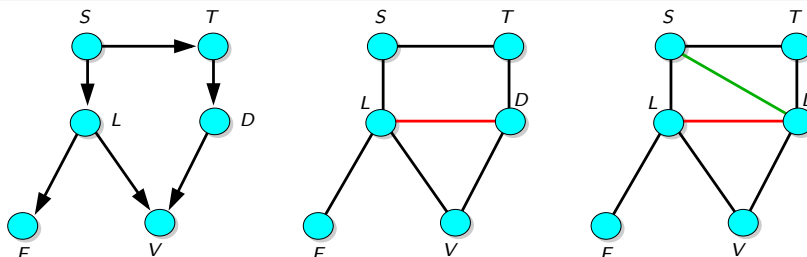
Construction de l'arbre de jonction (2)

L'existence de l'arbre de jonction n'est assuré que si le graphe est **triangulé** (c'est-à-dire : pas de cycle de longueur > 3 sans corde).

Triangulation

On obtient un graphe triangulé par **élimination des variables** :

- Supprimer itérativement tous les nœuds du graphe.
- Chaque suppression d'un nœud entraîne de relier tous ses voisins entre eux.
- Commencer par les nœuds n'appartenant qu'à une seule clique.



un ordre d'élimination : F, V, T, S, L, D

MADI

Inférences probabilistes dans les réseaux bayésiens

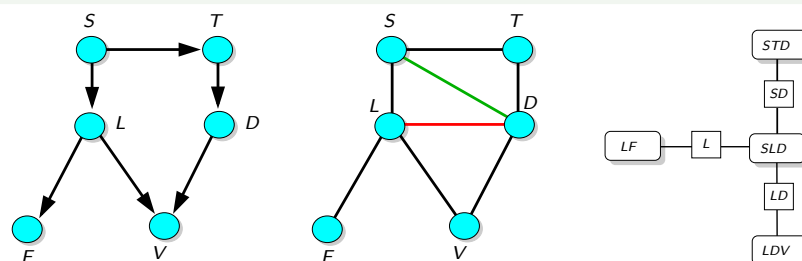
38 / 42

Construction de l'arbre de jonction (3)

L'arbre de jonction aura donc comme nœuds les cliques du graphe moralisé puis triangulé. Comment créer les arcs entre les cliques ?

Arbre de jonction

- Énumérer chaque clique du graphe triangulé.
- Pour chaque clique C_i , trouver la clique C_j ($j < i$) d'intersection maximale avec C_i .
- Relier C_i et C_j . Noter le séparateur $S_{ij} = C_i \cap C_j$.



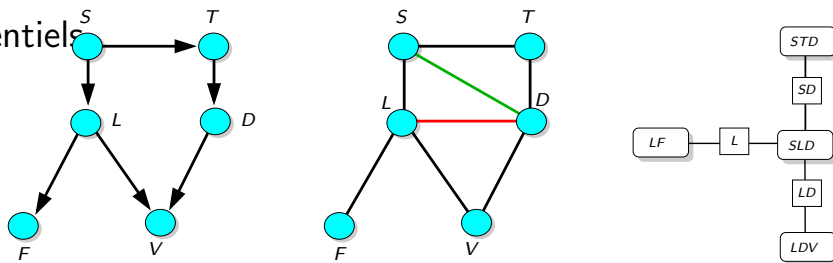
Il n'y a pas unicité de l'arbre de jonction pour un réseau bayésien.

MADI

Inférences probabilistes dans les réseaux bayésiens

39 / 42

Potentiel



Une décomposition de $P(V)$ suivant les cliques :

$$P(v) = \underbrace{P(s) \cdot P(l | s)}_{\phi_1(d, l, s)} \cdot \underbrace{P(t | s) \cdot P(d | t)}_{\phi_2(s, t, d)} \cdot \underbrace{P(f | l)}_{\phi_3(f, l)} \cdot \underbrace{P(v | l, d)}_{\phi_4(v, l, d)}$$

Une autre décomposition :

$$P(v) = \frac{P(s, t, d) \cdot P(s, l, d) \cdot P(l, f) \cdot P(l, d, v)}{P(s, d) \cdot P(l) \cdot P(l, d)}$$

Factorisation de P

$$P(v) = \frac{\prod_i \Phi_{C_i}(c_i)}{\prod_{i < j} \Phi_{S_{ij}}(s_{ij})}$$



Propagation sur les potentiels dans l'arbre de jonction

But : Fournir à toutes les cliques (C)/séparateurs (S) un potentiel qui soit la loi jointe de ses variables.

Propagation dans l'arbre de jonction

- **Initialisation :**
 $\forall C_i \in \mathcal{C}$, énumérées dans un ordre topologique,
 $\Psi_{C_i}^0 = \prod_{X \in C_i, X \notin C_j, j < i} P(X | \Pi_X)$ (et les observations associées)
 $\forall S \in \mathcal{S}$, $\Psi_S^0 = 1$ (fonction constante).
- **Collecte :** soit une clique C_j dont toutes les cliques adjacentes C_k sauf une unique C_i ont calculé leurs $\Psi_{C_k}^1$. Alors

$$\Psi_{S_{ij}}^1(s) = \sum_{C_i \setminus S_{ij}} \Psi_{C_i}^1(c) \quad \text{et} \quad \Psi_{C_j}^1 = \Psi_{C_j}^0 \cdot \frac{\Psi_{S_{ij}}^1}{\Psi_{S_{ij}}^0}$$
- **Distribution :** le dernier nœud de l'étape précédente, racine de la propagation, distribue vers tous ses voisins (qui feront de même) en utilisant exactement les mêmes formules que ci-dessus.

$$\Psi_{S_{ij}}^2(s) = \sum_{C_i \setminus S_{ij}} \Psi_{C_i}^2(c) \quad \text{et} \quad \Psi_{C_j}^2 = \Psi_{C_j}^1 \cdot \frac{\Psi_{S_{ij}}^2}{\Psi_{S_{ij}}^1}$$



Petit guide sur Jensen et Lazy Propagation

Quelques références

- 1 S.L. Lauritzen, D.J. Spiegelhalter (1988) *Local computations with probabilities on graphical structures and their application to expert systems(with discussion)*, Journal of the Royal Statistical Society, Series B, 50, pp.157–224.
- 2 F.V. Jensen, S.L. Lauritzen, K.G. Olesen (1990) *Bayesian Updating in Causal Probabilistic Networks by Local Computations*, Comp. Stat. Quarterly, 4, pp.269–282.
- 3 A.L. Madsen, F.V. Jensen (1998) *Lazy Propagation in Junction Trees*, Proceedings d'UAI-98.
- 4 A.L. Madsen, F.V. Jensen (1999) *Lazy Propagation : A Junction Tree Inference Algorithm Based on Lazy Evaluation*, Artificial Intelligence, 113, pp.203–245.

