

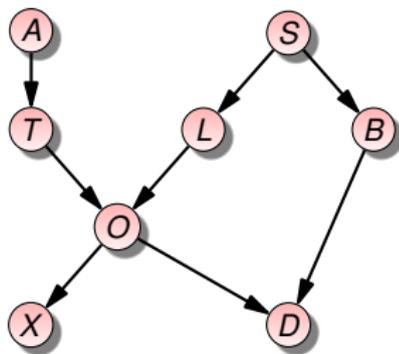
cours 4
Apprentissage de structure
de réseau bayésien

① Algorithme fondé sur les contraintes

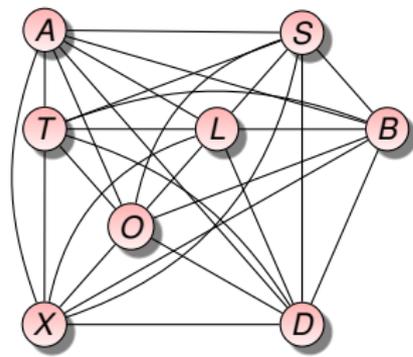
Apprentissage du squelette

- 1 $\mathcal{G} \leftarrow$ graphe non orienté complet
- 2 $d \leftarrow 0$ // taille de l'ensemble de conditionnement
- 3 $SepSet_{XY} \leftarrow \emptyset$ pour tout couple de nœuds X, Y
- 4 **répéter**
- 5 **pour chaque** *couple* (X, Y) t.q. $X - Y \in \mathcal{G}$ et $|\mathbf{Adj}(X) \setminus \{Y\}| \geq d$ **faire**
- 6 **répéter**
- 7 Choisir un $\mathbf{Z} \subseteq \mathbf{Adj}(X) \setminus \{Y\}$ t.q. $|\mathbf{Z}| = d$
- 8 **si** $X \perp\!\!\!\perp_P Y | \mathbf{Z}$ **alors**
- 9 Supprimer l'arête $X - Y$ de \mathcal{G}
- 10 $SepSet_{XY} \leftarrow \mathbf{Z}$
- 11 break
- 12 **jusqu'à** *Tous les* \mathbf{Z} de taille d ont été testés;
- 13 $d \leftarrow d + 1$
- 14 **jusqu'à** $|\mathbf{Adj}(X)| \leq d$ pour tout nœud X ;

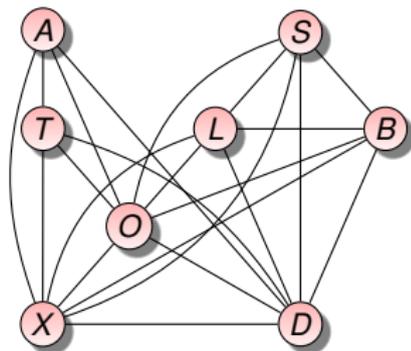
Exemple d'application (1/3)



RB réel



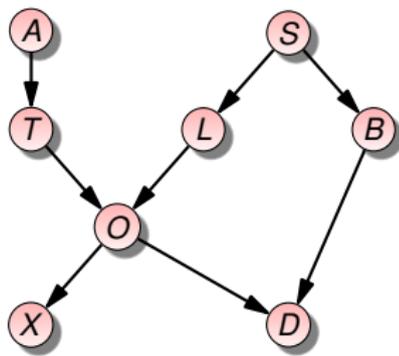
\mathcal{G} complet



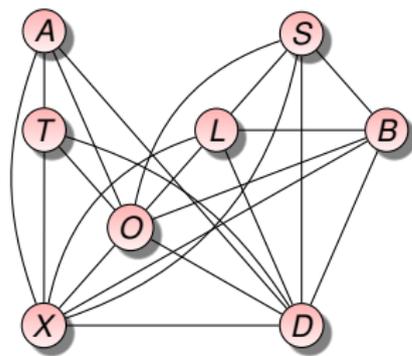
$\perp\!\!\!\perp$ ordre $d = 0$

- ▶ $A \perp\!\!\!\perp B, A \perp\!\!\!\perp L, A \perp\!\!\!\perp S$
- ▶ $B \perp\!\!\!\perp T$
- ▶ $L \perp\!\!\!\perp T$
- ▶ $S \perp\!\!\!\perp T$

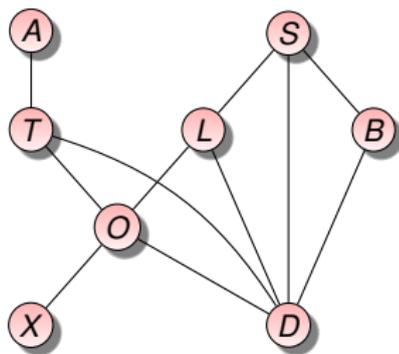
Exemple d'application (2/3)



RB réel



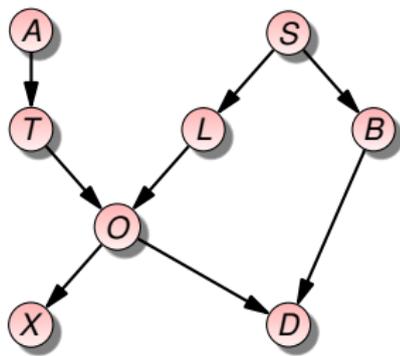
$\perp\!\!\!\perp$ ordre $d = 0$



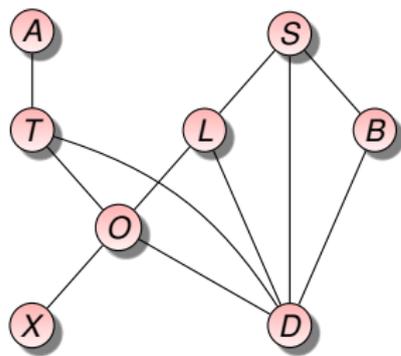
$\perp\!\!\!\perp$ ordre $d = 1$

- ▶ $A \perp\!\!\!\perp D \mid T, A \perp\!\!\!\perp O \mid T$
- ▶ $A \perp\!\!\!\perp X \mid T, B \perp\!\!\!\perp L \mid S$
- ▶ $B \perp\!\!\!\perp O \mid S, B \perp\!\!\!\perp X \mid S$
- ▶ $D \perp\!\!\!\perp X \mid O, L \perp\!\!\!\perp X \mid O$
- ▶ $O \perp\!\!\!\perp S \mid L, S \perp\!\!\!\perp X \mid L$
- ▶ $T \perp\!\!\!\perp X \mid O$

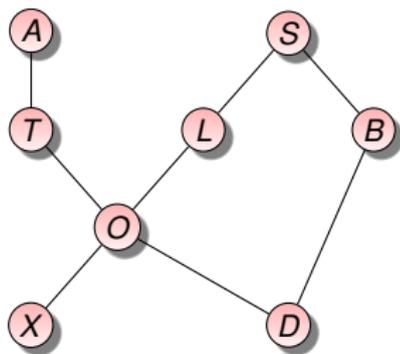
Exemple d'application (3/3)



RB réel



$\perp\!\!\!\perp$ ordre $d = 1$



$\perp\!\!\!\perp$ ordre $d = 2$

- ▶ $D \perp\!\!\!\perp L \mid \{O, B\}$
- ▶ $D \perp\!\!\!\perp S \mid \{O, B\}$
- ▶ $D \perp\!\!\!\perp T \mid \{O, S\}$

Définition

- ▶ **unshielded triple** : triplet $\langle X, Z, Y \rangle$ de nœuds de \mathcal{G} t.q. :
 - ▶ $X - Z - Y \in \mathcal{G}$
 - ▶ $X - Y \notin \mathcal{G}$
- ▶ équivalent non orienté d'une v-structure

Orientation des arêtes

1 // 1. Orientation des v-structures

2 $\mathcal{G}_{PDAG} \leftarrow \mathcal{G}$

3 **pour chaque unshielded triple** $\langle X, Z, Y \rangle$ **de** \mathcal{G} **faire**

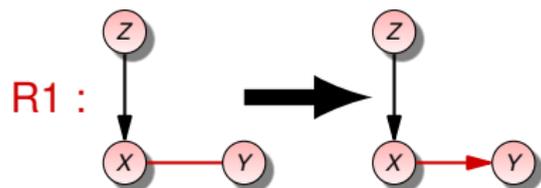
4 **si** $Z \notin \text{SepSet}_{XY}$ **alors**

5 **R0** : dans \mathcal{G}_{PDAG} , remplacer $X - Z$ et $Z - Y$ par $X \rightarrow Z$ et $Z \leftarrow Y$

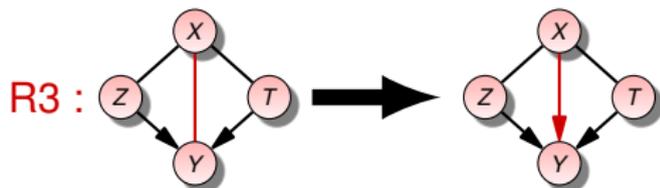
$\implies \mathcal{G}_{PDAG} = \text{squelette} + \text{v-structures} = \text{pattern}$

4 règles de propagation :

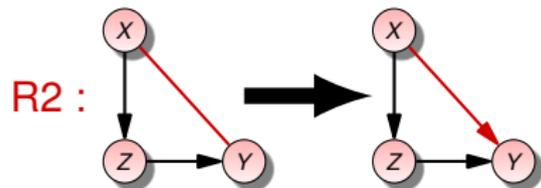
pas de v-structure :



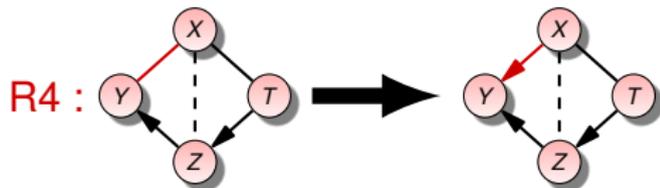
ni circuit ni v-structure :



pas de circuit :



ni circuit ni v-structure :



$$X \text{ --- } Y = X \text{ --- } Y \text{ OU } X \rightarrow Y \text{ OU } X \leftarrow Y$$

L'algorithme PC – Phase 2 : les propagations (2/2)

- ▶ CPDAG = completed partially directed acyclic graph
= représentant de la classe d'équivalence de Markov

Théorème – Orientation soundness – Meek(95)

- ▶ Les règles R1 à R4 sont sûres (soundness) :
pas respectées $\implies \exists$ nouvelle v-structure ou un circuit

Théorème – Orientation completeness – Meek(95)

- ▶ Appliquer R1, R2, R3 sur un pattern \implies CPDAG

Théorème – Completeness with background knowledge – Meek(95)

- ▶ Background knowledge \mathcal{K} : ensemble d'arcs interdits +
ensemble d'arcs obligatoires
- ▶ Appliquer R1 à R4 sur un pattern et orienter les arêtes
selon $\mathcal{K} \implies$ CPDAG

Orientation des arêtes

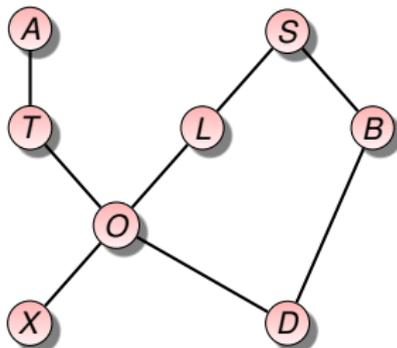
```
1 // Orientation des v-structures
2  $\mathcal{G}_{PDAG} \leftarrow \mathcal{G}$ 
3 pour chaque unshielded triple  $\langle X, Z, Y \rangle$  de  $\mathcal{G}$  faire
4   | si  $Z \notin \text{SepSet}_{XY}$  alors
5   |   | R0 : dans  $\mathcal{G}_{PDAG}$ , remplacer  $X - Z$  et  $Z - Y$  par  $X \rightarrow Z$  et  $Z \leftarrow Y$ 
6 // Propagations
7 répéter
8   | pour chaque arête  $X - Y \in \mathcal{G}_{PDAG}$  faire
9   |   | si  $(X - Y)$  arête rouge d'une règle  $R1, R2, R3$  ou  $R4$  alors
10  |   |   | Orienter l'arête selon la règle
11 jusqu'à ce que plus aucune arête ne puisse être orientée;
```

Exemple d'application (1/2)

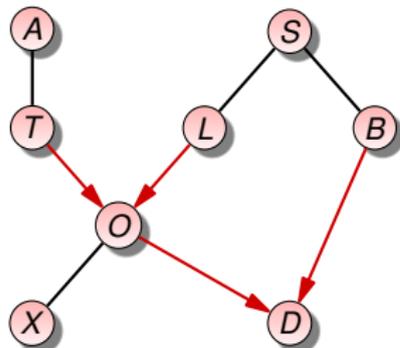
- 1 pour chaque *unshielded triple* $\langle X, Z, Y \rangle$ de \mathcal{G} faire
- 2 si $Z \notin \text{SepSet}_{XY}$ alors
- 3 $\left[\begin{array}{l} \text{R0 : dans } \mathcal{G}_{PDAG}, \text{ remplacer } X - Z \text{ et } Z - Y \text{ par } X \rightarrow Z \text{ et } Z \leftarrow Y \end{array} \right.$

Sepsets :

- ▶ $S_{AB} = S_{AL} = S_{AS} = \emptyset$ $S_{AD} = S_{AO} = S_{AX} = \{T\}$
- ▶ $S_{BT} = \emptyset$ $S_{BL} = S_{BO} = S_{BX} = \{S\}$
- ▶ $S_{DL} = S_{DS} = \{O, B\}$ $S_{DT} = \{O, S\}$ $S_{DX} = \{O\}$
- ▶ $S_{LT} = \emptyset$ $S_{LX} = \{O\}$ $S_{OS} = \{L\}$
- ▶ $S_{ST} = \emptyset$ $S_{SX} = \{L\}$ $S_{TX} = \{O\}$



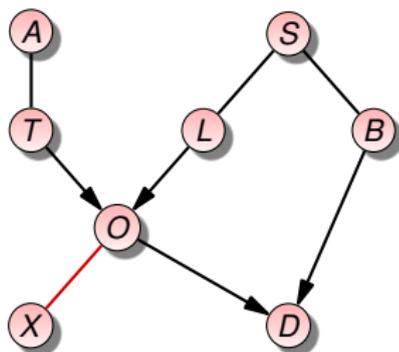
Après phase 1



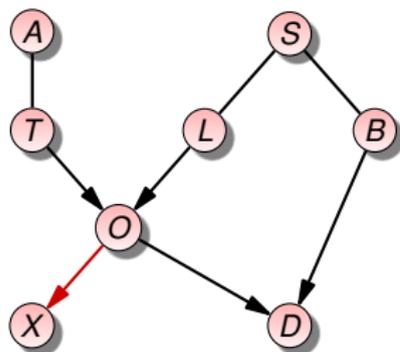
Après R0

Exemple d'application (2/2)

- 1 // Propagation sans rajouter de v-structure
- 2 **pour chaque** arête $(X, Y) \in \mathcal{G}_{PDAG}$ **faire**
- 3 **si** $(X - Y)$ arête rouge de $R1$ ($Z \rightarrow X$ et $X - Y$) **alors**
- 4 | remplacer $X - Y$ par $X \rightarrow Y$



Après $R0$



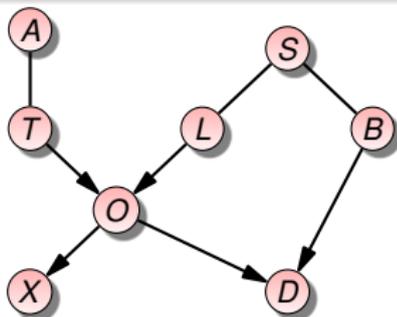
Après $R1$

Résultat : représentant de la classe d'équivalence de Markov

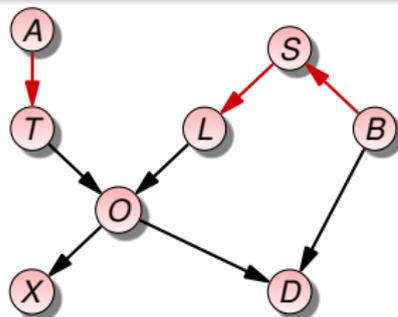
► CPDAG : completed partially directed acyclic graph

Phase 3 : compléter les orientations sans créer de v-structure

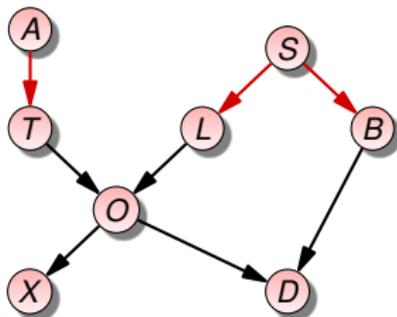
6 possibilités :



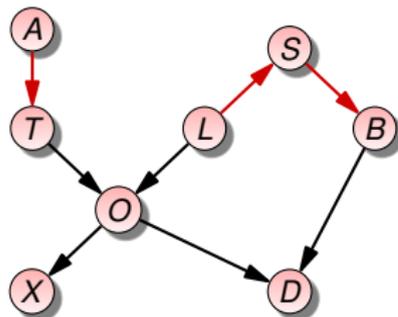
Après $R1$



1ère possibilité



2ème possibilité



3ème possibilité

► 3 autres possibilités similaires mais avec $T \rightarrow A$

Tests d'indépendance conditionnelle (1/2)

- ▶ X, Y : 2 variables aléatoires, \mathbf{Z} : ensemble de variables
- ▶ N_{xyz} : nombre d'occurrences de $(X = x, Y = y, \mathbf{Z} = \mathbf{z})$ dans \mathbf{D}
- ▶ $N_{xz} = \sum_{y \in \Omega_Y} N_{xyz}$ $N_{yz} = \sum_{x \in \Omega_X} N_{xyz}$ $N_{\mathbf{z}} = \sum_{x \in \Omega_X} N_{xz}$

Test du χ^2

$$\chi_{\text{statistics}}^2(X, Y|\mathbf{Z}) = \sum_{x \in \Omega_X} \sum_{y \in \Omega_Y} \sum_{\mathbf{z} \in \Omega_{\mathbf{Z}}} \frac{\left(N_{xyz} - \frac{N_{xz} N_{yz}}{N_{\mathbf{z}}} \right)^2}{\frac{N_{xz} N_{yz}}{N_{\mathbf{z}}}}$$

- ▶ Nb degrés de liberté : $df = (|\Omega_X| - 1) \times (|\Omega_Y| - 1) \times |\Omega_{\mathbf{Z}}|$
- ▶ α : niveau de risque (souvent 5%)
- ▶ Si $\chi_{\text{statistics}}^2(X, Y|\mathbf{Z}) \leq \chi^2(df, \alpha)$ alors $X \perp\!\!\!\perp Y|\mathbf{Z}$



Règle usuelle : ne faire le test que si $N_{xyz} \geq 5$ pour tout x, y, \mathbf{z}

Test du G^2

$$G^2_{statistics}(X, Y|Z) = 2 \sum_{x \in \Omega_X} \sum_{y \in \Omega_Y} \sum_{z \in \Omega_Z} N_{xyz} \ln \frac{N_{xyz} N_z}{N_{xz} N_{yz}}$$

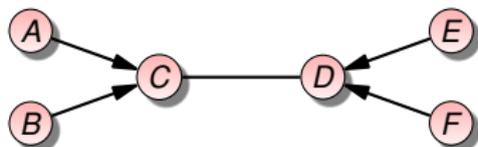
- ▶ Nb degrés de liberté : $df = (|\Omega_X| - 1) \times (|\Omega_Y| - 1) \times |\Omega_Z|$
- ▶ α : niveau de risque (e.g. 5%)
- ▶ Si $G^2_{statistics}(X, Y|Z) \leq \chi^2(df, \alpha)$ alors $X \perp\!\!\!\perp Y|Z$
- ▶ En pratique, tests du G^2 plus robustes que ceux du χ^2

- ▶ Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête  $X - Y$  t.q.  $|\text{Adj}(X) \setminus \{Y\}| \geq d$  faire
2   ...
3   Supprimer l'arête  $X - Y$  de  $\mathcal{G}$ 
4   ...
```

⇒ PC-stable, variante de [Colombo et Maathuis \(2014\)](#)

- ▶ $|\mathbf{D}|$ petite ⇒ tests χ^2 et G^2 peu fiables
- ▶ Phase 3 : pas d'orientation possible :



- ▶ cause 1 : pas de DAG-faithfulness (relations déterministes ?)
[Luo \(2006\)](#), [Rodrigues de Morais et al. \(2008\)](#), [Mabrouk et al. \(2014\)](#)
- ▶ cause 2 : erreurs dans les tests statistiques
- ▶ cause 3 : présence de variables latentes (non observées)
⇒ IC* [Verma \(1993\)](#), FCI [Spirtes, Glymour et Scheines \(2000\)](#)

② Apprentissage à base de scores



Meilleure structure : celle qui colle le mieux aux données

⇒ ① vraisemblance : $\mathcal{G}^* = \text{Argmax}_{\mathcal{G}} \mathcal{L}(\mathcal{G} : \mathbf{D})$

② \mathcal{G}^* = structure choisie avec le moins de « regret »

⇒ différents scores

Propriétés souhaitables des scores

▶ **Rasoir d'Occam :**

Privilégier les \mathcal{G} simples plutôt que complexes

▶ **Consistance locale :**

Ajouter un arc « utile » devrait augmenter le score

Ajouter un arc « inutile » devrait diminuer le score

▶ **Score équivalence :**

2 RB Markov-équivalents devraient avoir le même score

▶ **Décomposition locale :**

L'ajout/retrait d'un arc \implies score mis à jour en ne regardant que la partie de \mathcal{G} autour de l'arc

Voisinage d'un DAG \mathcal{G}

Voisinage de \mathcal{G} = ensemble des graphes \mathcal{G}' t.q. :

- ▶ \mathcal{G}' est un DAG
- ▶ \mathcal{G}' s'obtient en appliquant à \mathcal{G} un opérateur parmi :
 - ▶ l'ajout d'un arc
 - ▶ la suppression d'un arc
 - ▶ le retournement d'un arc

On note $\mathcal{N}(\mathcal{G})$ le voisinage de \mathcal{G}



- 1 Parcourir l'espace des DAG en partant d'un graphe \mathcal{G}_0
- 2 en trouvant dans son voisinage le graphe \mathcal{G}' de score plus élevé
- 3 en itérant le processus avec \mathcal{G}' jusqu'à un optimum (local)

⇒ Algorithme « Greedy Hill Climbing »

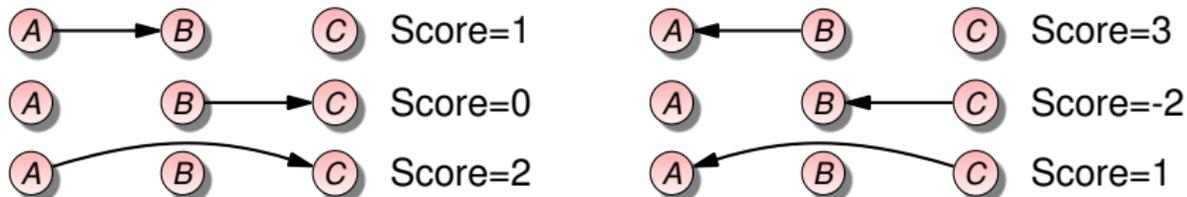
Algorithme d'apprentissage « glouton » :

```
1 // étape ①
2  $\mathcal{G}_{best} \leftarrow$  graphe orienté vide (sans arc) // meilleur graphe trouvé
3  $sc_{best} \leftarrow \text{Score}(\mathcal{G}_{best})$  // score du meilleur DAG
4
5 // parcours de l'espace des DAG – étape ③
6 répéter
7   // recherche du meilleur voisin
8    $\mathcal{N} \leftarrow$  voisinage de  $\mathcal{G}_{best}$  // calcul du voisinage
9   trouvé  $\leftarrow$  false // meilleur voisin pas encore trouvé
10  // parcours du voisinage – étape ②
11  pour chaque  $\mathcal{G}' \in \mathcal{N}$  faire
12     $sc' \leftarrow \text{Score}(\mathcal{G}')$ 
13    si  $sc' > sc_{best}$  alors
14       $\mathcal{G}_{best} \leftarrow \mathcal{G}'$ ,  $sc_{best} \leftarrow sc'$ 
15      trouvé  $\leftarrow$  true
16 jusqu'à trouvé = false;
17 Retourner  $\mathcal{G}_{best}$  // graphe localement optimal
```

Exemple d'application

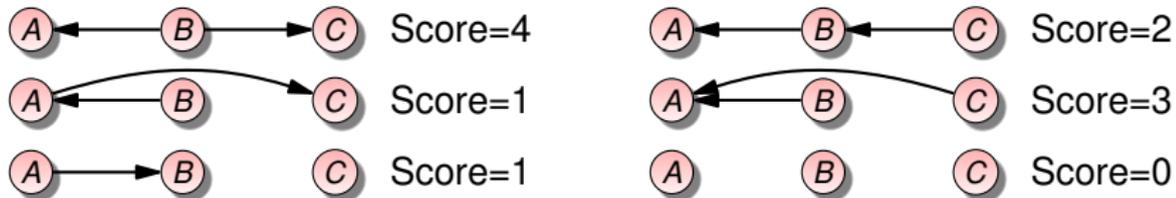
▶ Étape 1 : $\mathcal{G}_{best} = \textcircled{A} \quad \textcircled{B} \quad \textcircled{C} \quad SC_{best} = 0$

▶ Voisinage \mathcal{N} :



▶ Nouveau graphe $\mathcal{G}_{best} = \textcircled{A} \leftarrow \textcircled{B} \quad \textcircled{C} \quad SC_{best} = 3$

▶ Nouveau voisinage \mathcal{N} :



▶ Nouveau graphe $\mathcal{G}_{best} = \textcircled{A} \leftarrow \textcircled{B} \rightarrow \textcircled{C} \quad SC_{best} = 4$

▶ Nouveau voisinage \mathcal{N} :

.....

Score $BD(\mathcal{G}|\mathbf{D})$ [Heckerman, Geiger and Chickering (1995)]

$$\text{Score}_{BD}(\mathcal{G}|\mathbf{D}) = \prod_{i=1}^n \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(N_{ij} + \alpha_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(N_{ijk} + \alpha_{ijk})}{\Gamma(\alpha_{ijk})}$$

► $\Gamma(\cdot)$: généralisation continue de la factorielle

$$\Gamma(n) = (n - 1)! \text{ pour tout } n \in \mathbb{N}^*$$

► $\text{Score}_{BD}(X_i|\mathbf{Pa}(X_i), \mathbf{D}) = \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{ij})}{\Gamma(N_{ij} + \alpha_{ij})} \prod_{k=1}^{r_i} \frac{\Gamma(N_{ijk} + \alpha_{ijk})}{\Gamma(\alpha_{ijk})}$

► $\text{Score}_{BD}(\mathcal{G}|\mathbf{D}) = \prod_{i=1}^n \text{Score}_{BD}(X_i|\mathbf{Pa}(X_i), \mathbf{D})$



En pratique, on calcule plutôt $\log(\text{Score}_{BD}(\mathcal{G}|\mathbf{D}))$

Score $BD(\mathcal{G}|\mathbf{D})$ [Heckerman, Geiger and Chickering (1995)]

▶ $\log(\text{Score}_{BD}(X_i|\mathbf{Pa}(X_i), \mathbf{D}))$

$$= \sum_{j=1}^{q_i} \log \Gamma(\alpha_{ij}) - \log \Gamma(N_{ij} + \alpha_{ij}) + \sum_{k=1}^{r_i} \log \Gamma(N_{ijk} + \alpha_{ijk}) - \log \Gamma(\alpha_{ijk})$$

▶ $\log(\text{Score}_{BD}(\mathcal{G}|\mathbf{D})) = \sum_{i=1}^n \log(\text{Score}_{BD}(X_i|\mathbf{Pa}(X_i), \mathbf{D}))$

Score K2

[Cooper et Herskovits (1992)]

► $\text{Score}_{K2}(\mathcal{G}|\mathbf{D}) = \log(\text{Score}_{BD}(\mathcal{G}|\mathbf{D}))$ avec :

$\alpha_{ijk} = 1$ pour tout i, j, k

⇒ revient à rajouter 1 dans chaque cellule des tableaux de contingence (de comptage)

⇒ revient à rajouter $r_i q_i$ observations dans \mathbf{D}

► $\text{Score}_{K2}(X_i|\mathbf{Pa}(X_i), \mathbf{D}) = \sum_{j=1}^{q_i} \left[\log \left(\frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \right) + \sum_{k=1}^{r_i} \log(N_{ijk}!) \right]$

► $\text{Score}_{K2}(\mathcal{G}|\mathbf{D}) = \sum_{i=1}^n \text{Score}_{K2}(X_i|\mathbf{Pa}(X_i), \mathbf{D})$

Exemple d'application (1/2)

$$\text{Score}_{K_2}(X_i | \mathbf{Pa}(X_i), \mathbf{D}) = \sum_{j=1}^{q_i} \left[\log \left(\frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \right) + \sum_{k=1}^{r_i} \log(N_{ijk}!) \right]$$

A

B

C

► $\mathbf{D} \implies r_i = 2$ pour tout i ($|\Omega_A| = |\Omega_B| = |\Omega_C| = 2$)

► $\mathcal{G} \implies q_i = 1$ (pas de parent) et $N_{ij} = |\mathbf{D}| = 7 \forall i$

► $i = 1$ (A) :

a_1	a_2
3	4

 $r_1 = |\Omega_A| = 2$
 $\mathbf{Pa}(A) = \emptyset \implies q_1 = 1$

$$\text{Score}_{K_2}(A) = \log \left(\frac{(2-1)!}{(7+2-1)!} \right) + \log(3!) + \log(4!) \approx -5,6348$$

► $i = 2$ (B) :

b_1	b_2
3	4

 $\text{Score}_{K_2}(B) \approx -5,6348$

► $i = 3$ (C) :

c_1	c_2
4	3

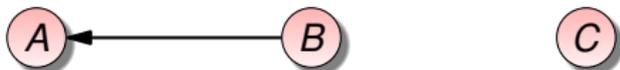
 $\text{Score}_{K_2}(C) \approx -5,6348$

► $\text{Score}_{K_2}(\mathcal{G}) \approx -5,6348 - 5,6348 - 5,6348 = -16,9044$

A	B	C
a_1	b_1	c_1
a_1	b_1	c_2
a_1	b_2	c_1
a_2	b_2	c_2
a_2	b_2	c_1
a_2	b_2	c_2
a_2	b_1	c_1

Exemple d'application (2/2)

$$\text{Score}_{K_2}(X_i | \mathbf{Pa}(X_i), \mathbf{D}) = \sum_{j=1}^{q_i} \left[\log \left(\frac{(r_i - 1)!}{(N_{ij} + r_i - 1)!} \right) + \sum_{k=1}^{r_i} \log(N_{ijk}!) \right]$$



- $\text{Score}_{K_2}(B)$ et $\text{Score}_{K_2}(C)$ inchangés $\approx -5,6348$

A	B	C
a_1	b_1	c_1
a_1	b_1	c_2
a_1	b_2	c_1
a_2	b_2	c_2
a_2	b_2	c_1
a_2	b_2	c_2
a_2	b_1	c_1

- $(A|B)$:

	a_1	a_2	N_{ij}
b_1	2	1	3
b_2	1	3	4

 $N_{ij} = \sum$ sur chaque ligne

$$\begin{aligned} \text{Score}_{K_2}(A|B) &= \log \left(\frac{(2-1)!}{(3+2-1)!} \right) + \log(2!) + \log(1!) \quad (\text{ligne } b_1) \\ &\quad + \log \left(\frac{(2-1)!}{(4+2-1)!} \right) + \log(1!) + \log(3!) \quad (\text{ligne } b_2) \\ &\approx -6,1738 \end{aligned}$$

- $\text{Score}_{K_2}(\mathcal{G}) \approx -6,1738 - 5,6348 - 5,6348 = -17,4434$
 \implies Graphe moins probable que celui du slide précédent

- ▶ **Problème** : Score K2 non score-équivalent :
2 RB Markov-équivalents n'ont pas forcément le même score !

Score BDeu

[Buntine (1991)]

- ▶ $\text{Score}_{BDeu}(\mathcal{G}|\mathbf{D}) = \log(\text{Score}_{BD}(\mathcal{G}|\mathbf{D}))$ avec :

$$\alpha_{ijk} = \frac{N'}{r_i q_i} \text{ pour tout } i, j, k \quad N' = \ll \text{effective sample size} \gg$$

⇒ revient à rajouter $\frac{N'}{r_i q_i}$ dans chaque cellule des tableaux de contingence (de comptage)

⇒ revient à rajouter N' observations dans \mathbf{D}

- ▶ $\text{Score}_{BDeu}(X_i | \mathbf{Pa}(X_i), \mathbf{D})$

$$= \sum_{j=1}^{q_i} \log \Gamma \left(\frac{N'}{q_i} \right) - \log \Gamma \left(N_{ij} + \frac{N'}{q_i} \right) + \sum_{k=1}^{r_i} \log \Gamma \left(N_{ijk} + \frac{N'}{r_i q_i} \right) - \log \Gamma \left(\frac{N'}{r_i q_i} \right)$$

- ▶ $\text{Score}_{BDeu}(\mathcal{G}|\mathbf{D}) = \sum_{i=1}^n \text{Score}_{BDeu}(X_i | \mathbf{Pa}(X_i), \mathbf{D})$

- ▶ BDeu : score équivalent [Heckerman, Geiger, Chickering (1995)]

▶ Score $BD(\mathcal{G}) : \int_{\Theta} P(\mathbf{D}|\mathcal{G}, \Theta)\pi(\Theta|\mathcal{G})d\Theta$

⇒ vraisemblance moyenne sur {RB de structure \mathcal{G} }

⇒ moyenne sur tous les paramètres

▶ Et si on utilisait les paramètres optimaux plutôt que la moyenne ?

Score(\mathcal{G}) : $\max_{\Theta} P(\mathbf{D}|\mathcal{G}, \Theta)$

⇒ Paramètres qui « collent » le plus aux données

⇒ Apprentissage de paramètres par max de vraisemblance

Fondement des scores issus de la théorie de l'information

- ▶ Optimisation des paramètres de \mathcal{G} par max de vraisemblance
⇒ score « log-likelihood »

Score log-likelihood (LL)

- ▶ $\text{Score}_{LL}(\mathcal{G}|\mathbf{D}) = \sum_{i=1}^n \text{Score}_{LL}(X_i|\mathbf{Pa}(X_i), \mathbf{D})$

- ▶ $\text{Score}_{LL}(X_i|\mathbf{Pa}(X_i), \mathbf{D}) = \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log \left(\frac{N_{ijk}}{N_{ij}} \right)$



En pratique, apprend des graphes trop denses

⇒ « Sur-apprentissage »

⇒ Rajouter une pénalité s'il y a trop d'arcs

Score MDL (minimum description length)

▶ $\text{Score}_{MDL}(\mathcal{G}|\mathbf{D}) = \text{Score}_{LL}(\mathcal{G}|\mathbf{D}) - \frac{1}{2} \log(N)|\mathcal{G}|$

▶ $|\mathcal{G}| = \sum_{i=1}^n (r_i - 1) \times q_i$: nb de paramètres θ_{ijk} à choisir

▶ $-\frac{1}{2} \log(N)|\mathcal{G}|$: pénalité

Permet de minimiser la taille mémoire pour stocker le RB

Score BIC (Bayesian Information Criterion)

▶ S'appuie sur le critère BIC [Schwarz (1978)]

▶ $\text{Score}_{BIC}(\mathcal{G}|\mathbf{D}) = \text{Score}_{MDL}(\mathcal{G}|\mathbf{D})$

Score AIC (Akaike Information Criterion)

▶ S'appuie sur le critère d'information d'Akaike (1973)

▶ $\text{Score}_{AIC}(\mathcal{G}|\mathbf{D}) = \text{Score}_{LL}(\mathcal{G}|\mathbf{D}) - |\mathcal{G}|$

Exemple d'application (1/2)

$$\text{Score}_{MDL}(\mathcal{G}|\mathbf{D}) = \text{Score}_{LL}(\mathcal{G}|\mathbf{D}) - \frac{1}{2} \log(N)|\mathcal{G}| \quad \text{Score}_{LL}(\mathcal{G}|\mathbf{D}) = \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} N_{ijk} \log \left(\frac{N_{ijk}}{N_{ij}} \right)$$

A

B

C

► $\mathbf{D} \implies r_i = 2, \mathcal{G} \implies q_i = 1$ (pas de parent)

A	B	C
a_1	b_1	c_1
a_1	b_1	c_2
a_1	b_2	c_1
a_2	b_2	c_2
a_2	b_2	c_1
a_2	b_2	c_2
a_2	b_1	c_1

► $i = 1$ (A) :

a_1	a_2	N_{ij}
3	4	7

 $N_{ij} = \text{total ligne} = 3 + 4$

$$\text{Score}_{LL}(A) = 3 \log \left(\frac{3}{7} \right) + 4 \log \left(\frac{4}{7} \right) \approx -4,78$$

► $i = 2$ (B) :

b_1	b_2
3	4

 $\text{Score}_{LL}(B) \approx -4,78$

► $i = 3$ (C) :

c_1	c_2
4	3

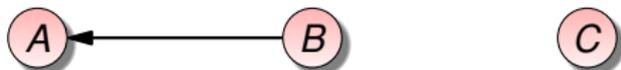
 $\text{Score}_{LL}(C) \approx -4,78$

► $|\mathcal{G}| = \sum_{i=1}^3 (r_i - 1) \times q_i = (1 \times 1) + (1 \times 1) + (1 \times 1) = 3$

► $\text{Score}_{MDL}(\mathcal{G}) \approx -4,78 \times 3 - 0,5 \times \log(7) \times 3 \approx -17,259$

Exemple d'application (2/2)

$$\text{Score}_{MDL}(X_i | \text{Pa}(X_i), \mathbf{D}) = \sum_{k=1}^{r_i} \left[N_{ijk} \log \left(\frac{N_{ijk}}{N_{ij}} \right) \right] - \frac{1}{2} \log(N) \times (r_i - 1) \times q_i$$



► $\text{Score}_{MDL}(B)$ et $\text{Score}_{MDL}(C) \approx -4,78 - \frac{1}{2} \log(7) \approx -5.753$

A	B	C
a ₁	b ₁	c ₁
a ₁	b ₁	c ₂
a ₁	b ₂	c ₁
a ₂	b ₂	c ₂
a ₂	b ₂	c ₁
a ₂	b ₂	c ₂
a ₂	b ₁	c ₁

► (A|B) :

	a ₁	a ₂	N _{ij}
b ₁	2	1	3
b ₂	1	3	4

$N_{ij} = \sum$ sur chaque ligne

$$\begin{aligned} \text{Score}_{MDL}(A|B) &= 2 \times \log \left(\frac{2}{3} \right) + 1 \times \log \left(\frac{1}{3} \right) && \text{(ligne } b_1) \\ &+ 1 \times \log \left(\frac{1}{4} \right) + 3 \times \log \left(\frac{3}{4} \right) && \text{(ligne } b_2) \\ &- \frac{1}{2} \log(7) \times 1 \times 2 \approx -6,105 && \text{(pénalité)} \end{aligned}$$

► $\text{Score}_{MDL}(\mathcal{G}) \approx -6,105 - 5.753 - 5.753 = -17,611$

⇒ Graphe moins probable que celui du slide précédent

- ▶ **Akaike, H. (1973)** « Information theory and an extension of the maximum likelihood principle », Proceedings of the 2nd International Symposium on Information Theory, 267–281
- ▶ **Buntine W. (1991)** « Theory refinement on Bayesian networks », Proceedings of Uncertainty in Artificial Intelligence, 52–60
- ▶ **Colombo D. et Maathuis M.H. (2014)** « Order-Independent Constraint-Based Causal Structure Learning », Journal of Machine Learning Research, 15 :3921–3962
- ▶ **Geiger D. et Heckerman D. (1997)** « A Characterization of the Dirichlet Distribution through Global and Local Parameter Independence », The Annals of Statistics, 25(3) :1344–1369
- ▶ **Heckerman D., Geiger D. et Chickering D. (1995)** « Learning Bayesian Networks : The Combination of Knowledge and Statistical Data », Machine Learning, 20 :197–243

- ▶ **Luo W. (2006)** « Learning Bayesian networks in semi-deterministic systems », Proceedings of the Canadian Conference on Artificial Intelligence, 230–241
- ▶ **Mabrouk A., Gonzales C., Jabet-Chevalier K. et Chojnaki E. (2014)** « An Efficient Bayesian Network Structure Learning Algorithm in the Presence of Deterministic Relations », Proceedings of the European Conference on Artificial Intelligence, 567–572
- ▶ **Meek C. (1995)** « Causal inference and causal explanation with background knowledge », Proceedings of the Conference on Uncertainty in Artificial Intelligence, 403–410
- ▶ **Pearl J. et Verma T. (1991)** « A theory of inferred causation », Proceedings of the 2nd International Conference on Knowledge Representation and Reasoning, 441–452

- ▶ **Rodrigues de Morais, S. Aussem, A. et Corbex M. (2008)** « Handling almost-deterministic relationships in constraint-based Bayesian network discovery : Application to cancer risk factor identification », Proceedings of the European Symposium on Artificial Neural Networks, 101–106
- ▶ **Schwarz, G.E. (1978)** « Estimating the dimension of a model », Annals of Statistics, 6(2) :461–464
- ▶ **Spirtes E., Glymour C. et Scheines R. (2000)** Causation, Prediction and Search, 2nd edition, Springer-Verlag
- ▶ **Verma T. (1993)** « Graphical aspects of causal models », Technical report R-191, UCLA, Computer Science Department