cours 4
Apprentissage fondé sur les contraintes

Aix*Marseille université
Initiative d'excellence

Master SID — Raisonnement dans l'incertain

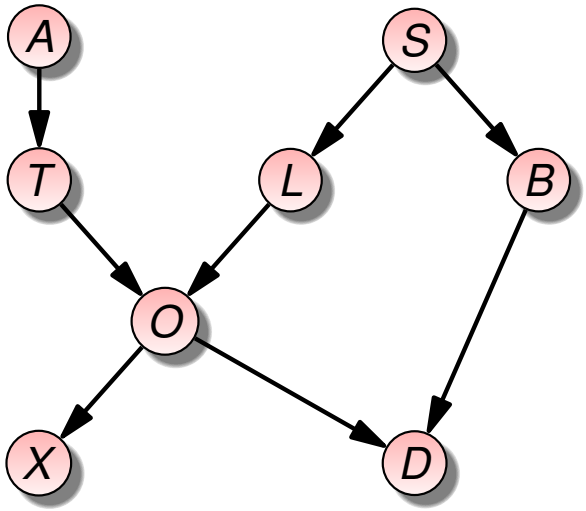
©CG(2026)

L'algorithme PC – Phase 1 : le squelette

Apprentissage du squelette

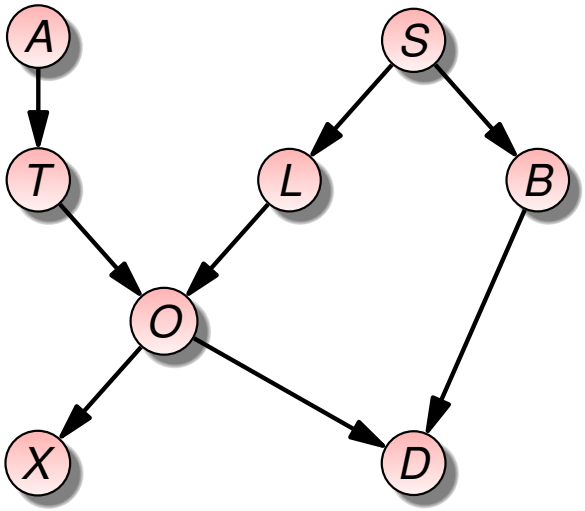
1 $\mathcal{G} \leftarrow$ graphe non orienté complet
2 $d \leftarrow 0$ // taille de l'ensemble de conditionnement
3 $SepSet_{XY} \leftarrow \emptyset$ pour tout couple de nœuds X, Y
4 **répéter**
5 **pour chaque** *couple* (X, Y) *t.q.* $X - Y \in \mathcal{G}$ *et* $|\mathbf{Adj}(X) \backslash \{Y\}| \geq d$ **faire**
6 **répéter**
7 Choisir un $\mathbf{Z} \subseteq \mathbf{Adj}(X) \backslash \{Y\}$ 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)

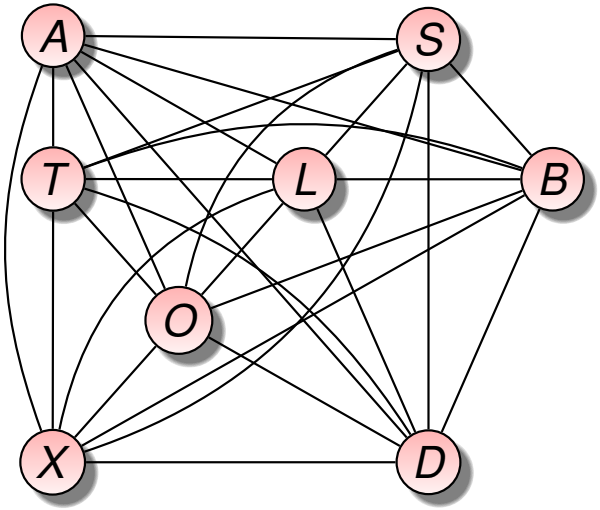


RB réel

Exemple d'application (1/3)

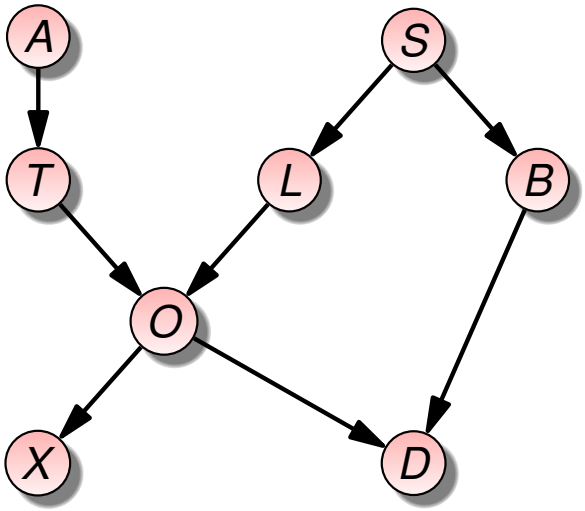


RB réel

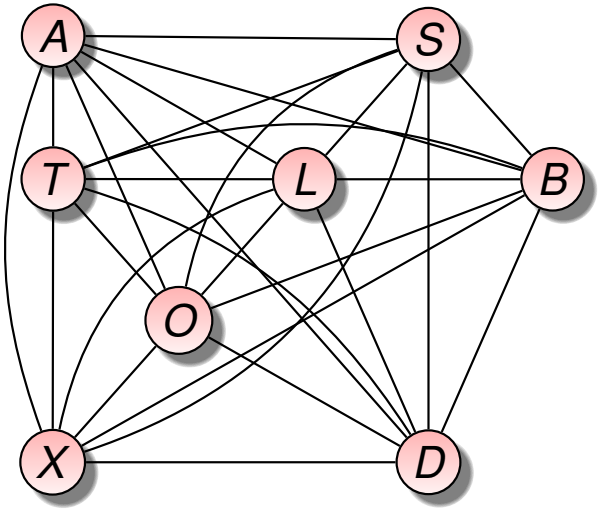


$\mathcal{G}$ complet

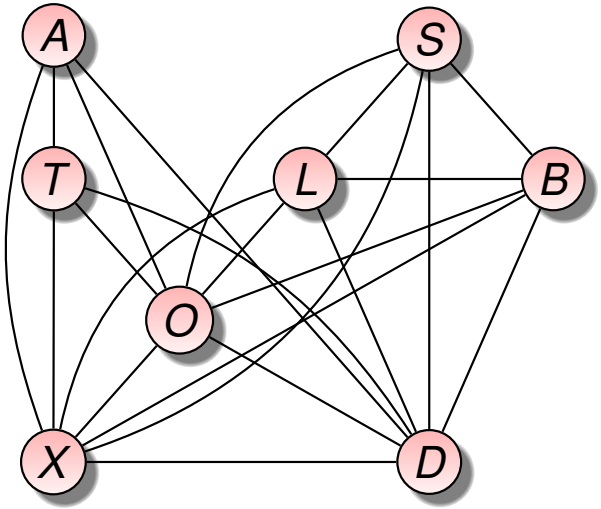
Exemple d'application (1/3)



RB réel



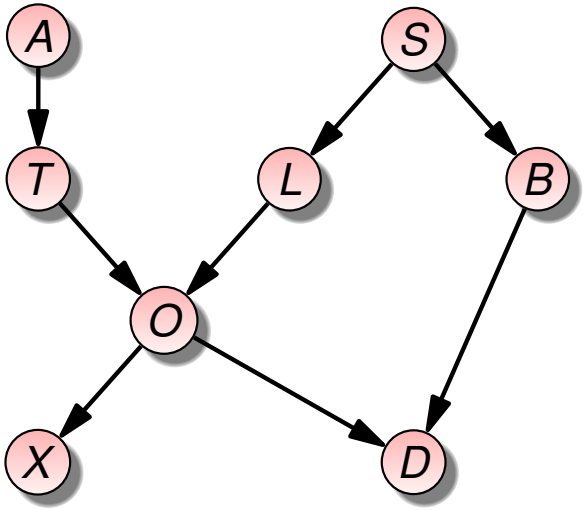
$\mathcal{G}$ complet



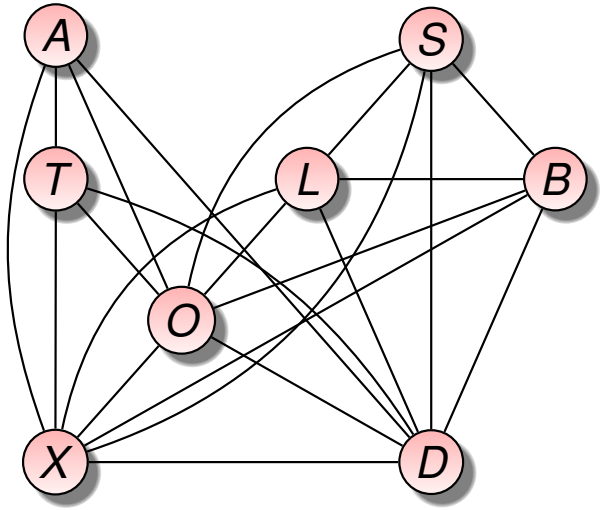
$\perp\!\!\!\perp$ ordre $d = 0$

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

Exemple d'application (2/3)

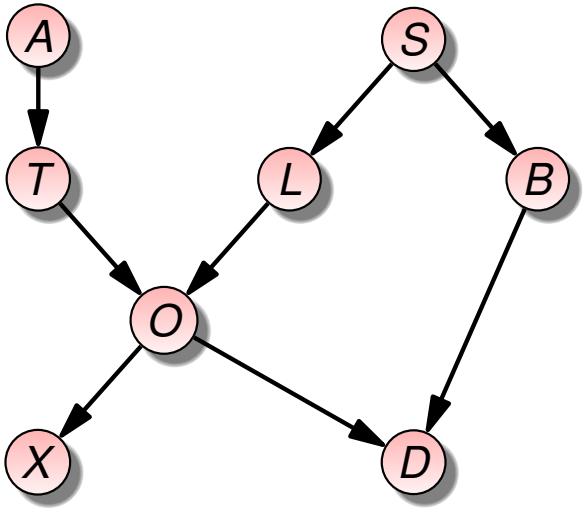


RB réel

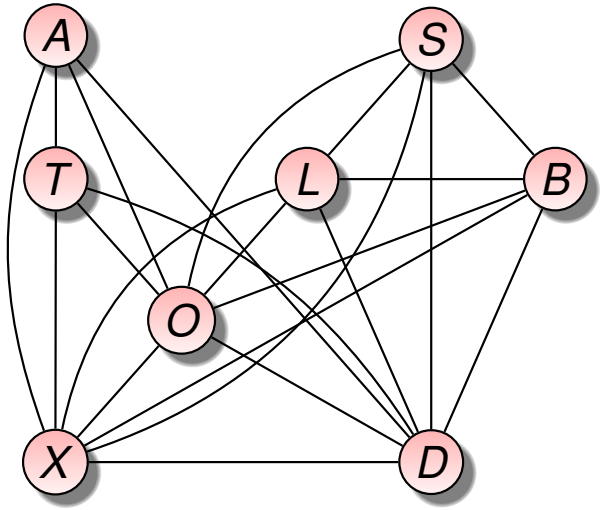


$\perp\!\!\!\perp$ ordre $d = 0$

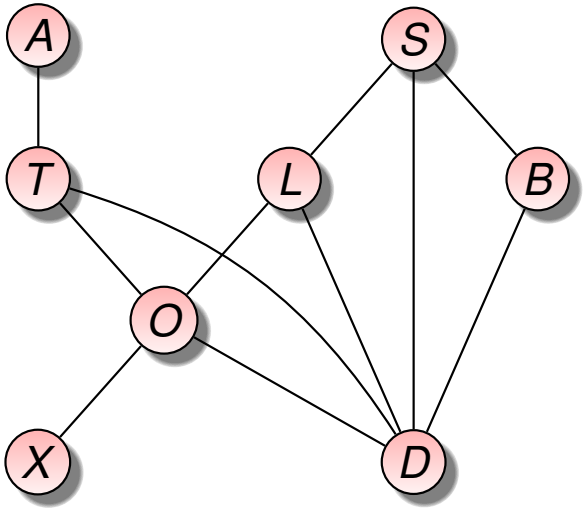
Exemple d'application (2/3)



RB réel



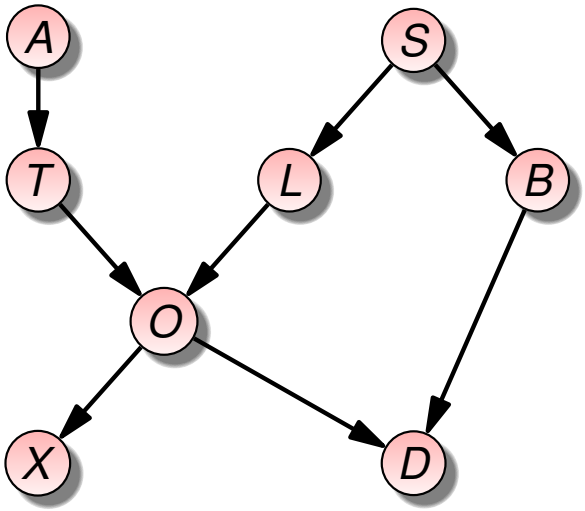
$\perp\!\!\!\perp$ ordre $d = 0$



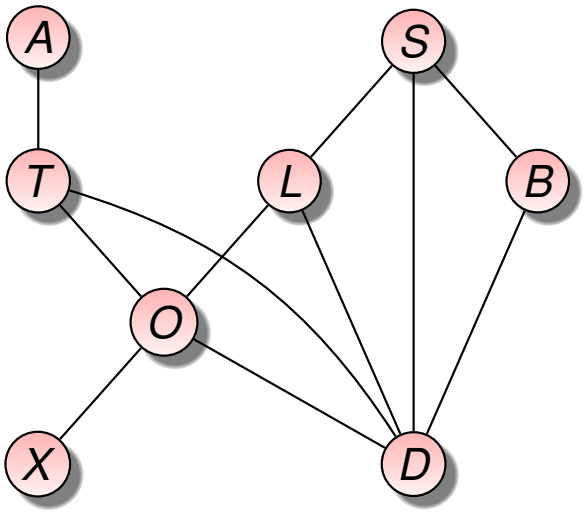
$\perp\!\!\!\perp$ ordre $d = 1$

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

Exemple d'application (3/3)

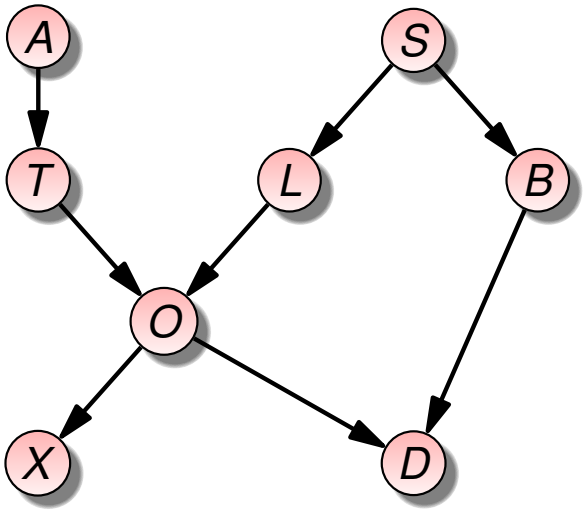


RB réel

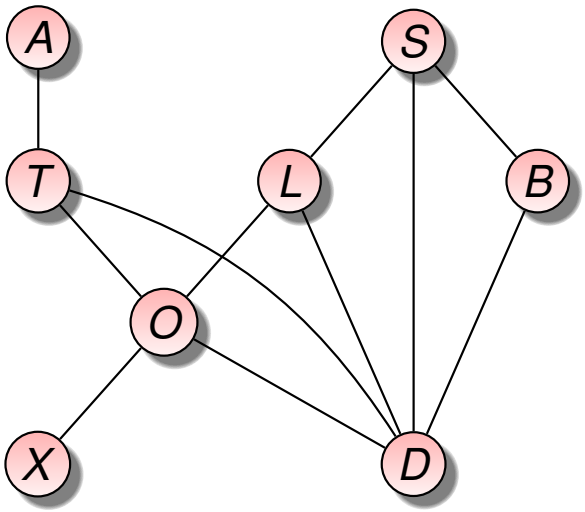


$\perp\!\!\!\perp$ ordre $d = 1$

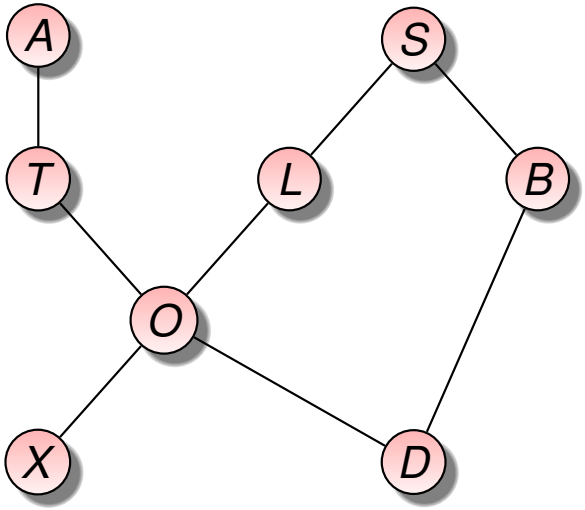
Exemple d'application (3/3)



RB réel



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



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

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

L'algorithme PC – Phase 2 : l'orientation (1/2)

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

L'algorithme PC – Phase 2 : l'orientation (1/2)

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

L'algorithme PC – Phase 2 : l'orientation (1/2)

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

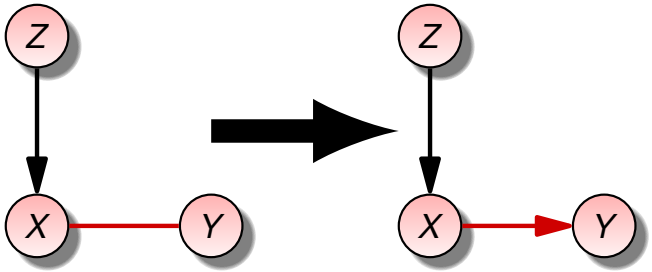
$\Longrightarrow \mathcal{G}_{PDAG} =$ squelette $+$ v-structures = pattern

L'algorithme PC – Phase 2 : les propagations (1/2)

4 règles de propagation :

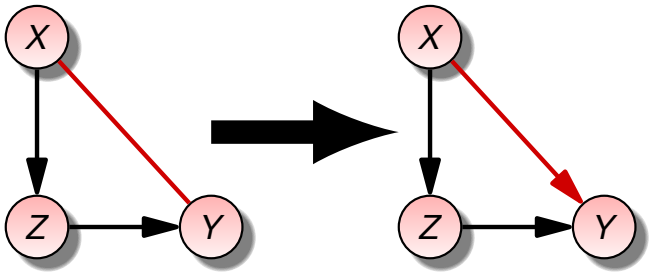
pas de v-structure :

R1 :



pas de circuit :

R2 :

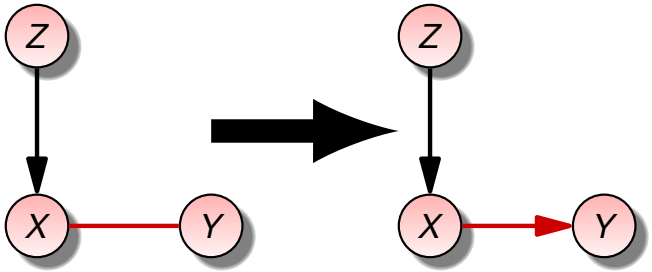


L'algorithme PC – Phase 2 : les propagations (1/2)

4 règles de propagation :

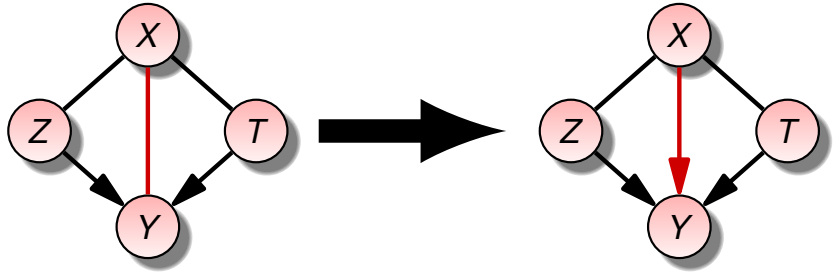
pas de v-structure :

R1 :



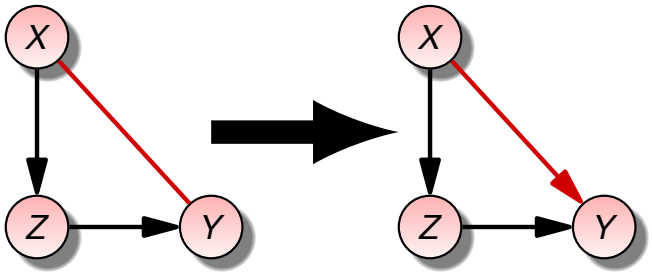
ni circuit ni v-structure :

R3 :



pas de circuit :

R2 :

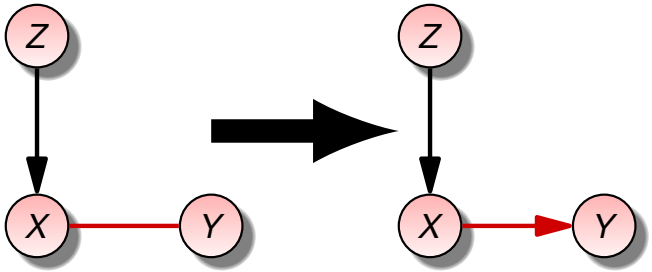


L'algorithme PC – Phase 2 : les propagations (1/2)

4 règles de propagation :

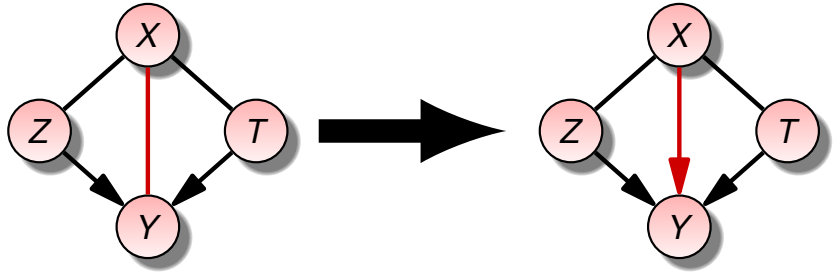
pas de v-structure :

R1 :



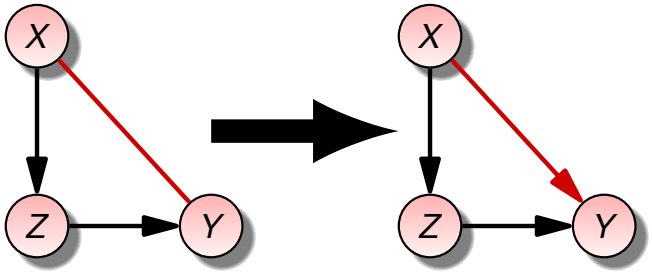
ni circuit ni v-structure :

R3 :



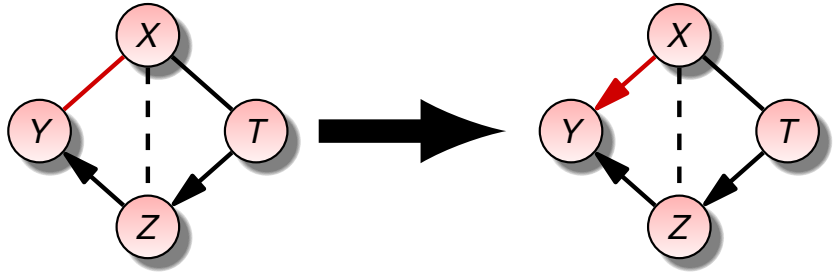
pas de circuit :

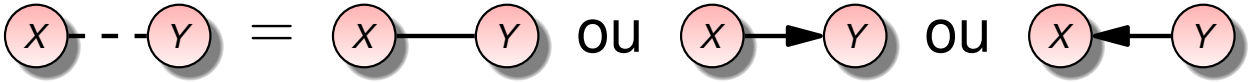
R2 :



ni circuit ni v-structure :

R4 :





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

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

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 $\Longrightarrow \exists$ nouvelle v-structure ou un circuit

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 $\Longrightarrow \exists$ nouvelle v-structure ou un circuit

Théorème – Orientation completeness – Meek(95)

- Appliquer R1, R2, R3 sur un pattern $\Longrightarrow$ CPDAG

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 $\Longrightarrow \exists$ nouvelle v-structure ou un circuit

Théorème – Orientation completeness – Meek(95)

- Appliquer R1, R2, R3 sur un pattern $\Longrightarrow$ CPDAG

Théorème – Completeness with background knowledge – Meek(95)

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

L'algorithme PC – Phase 2 : l'orientation (2/2)

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

Orientation des arêtes

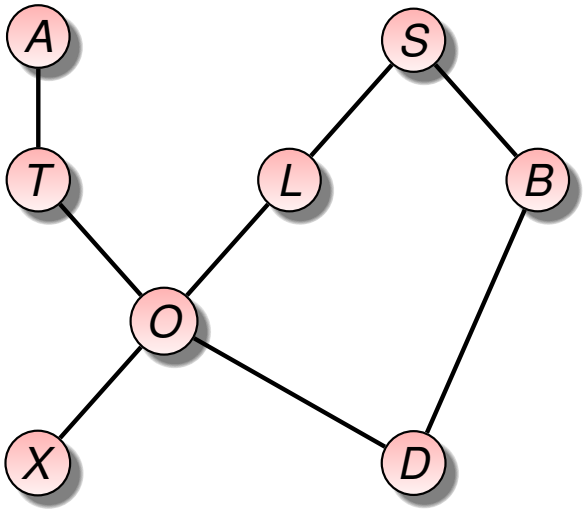
1 // Orientation des v-structures
2 $\mathcal{G}_{PDAG} \leftarrow \mathcal{G}$
3 **pour chaque** *unshielded triple* $\langle X, Z, Y \rangle$ *de* $\mathcal{G}$ **faire**
4 **si** $Z \notin SepSet_{XY}$ **alors**
5 *R*0 : dans $\mathcal{G}_{PDAG}$, remplacer $X - Z$ et $Z - Y$ par $X \rightarrow Z$ 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)

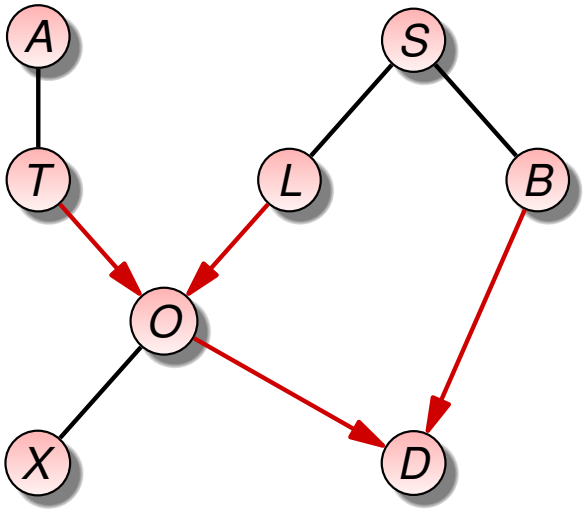
1 **pour chaque** *unshielded triple* $\langle X, Z, Y \rangle$ *de* $\mathcal{G}$ **faire**
2 **si** $Z \notin SepSet_{XY}$ **alors**
3 $R0$: dans $\mathcal{G}_{PDAG}$, remplacer $X - Z$ et $Z - Y$ par $X \rightarrow Z$ et $Z \leftarrow Y$

Sepsets :

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



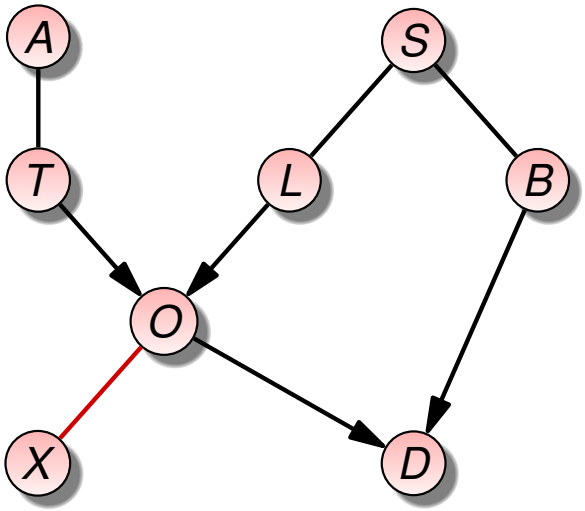
Après phase 1



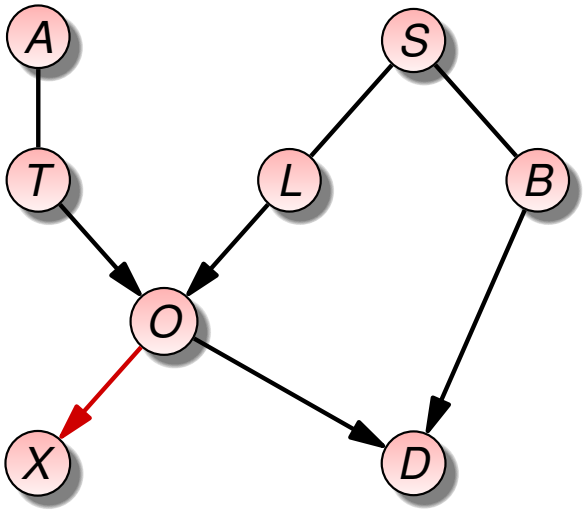
Après $R0$

Exemple d'application (2/2)

```
1 // Propagation sans rajouter de v-structure
2 pour chaque arête (X, Y) ∈ G_PDAG faire
3     si (X − Y) arête rouge de R1 (Z → X et X − Y) alors
4         remplacer X − Y par X → Y
```



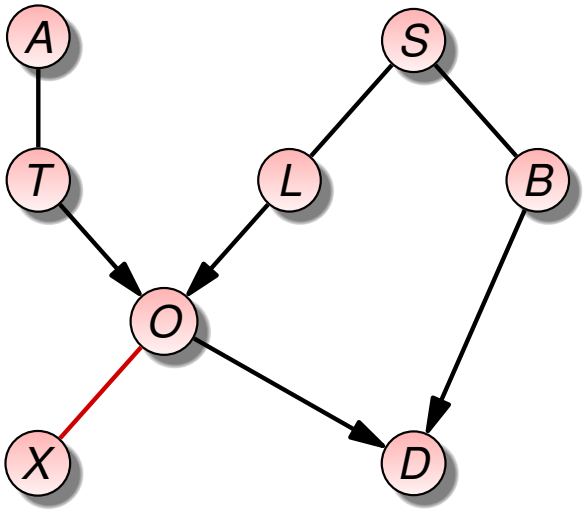
Après $R0$



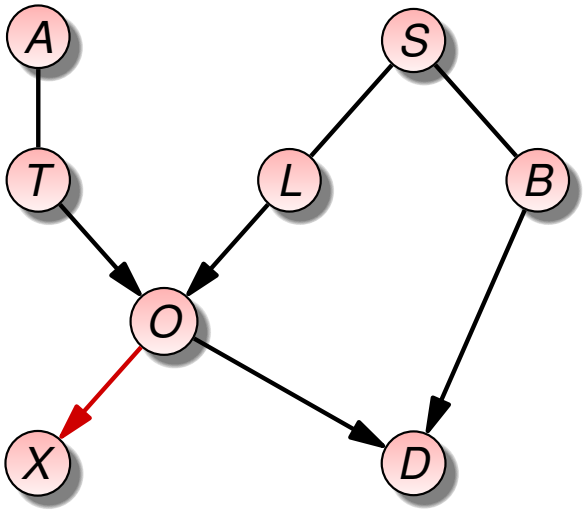
Après $R1$

Exemple d'application (2/2)

```
1 // Propagation sans rajouter de v-structure
2 pour chaque arête (X, Y) ∈ G_PDAG faire
3     si (X − Y) arête rouge de R1 (Z → X et X − Y) alors
4         remplacer X − Y par X → Y
```



Après $R0$

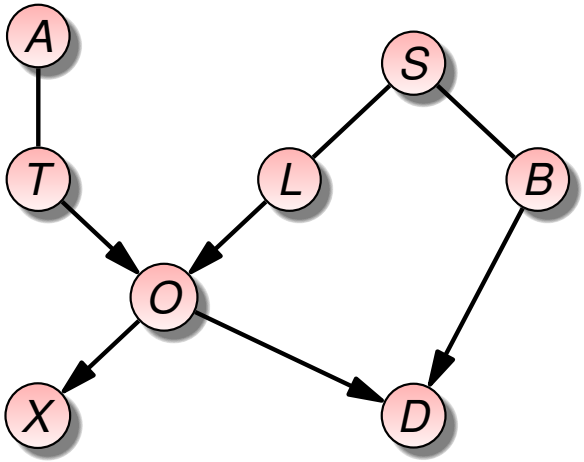


Après $R1$

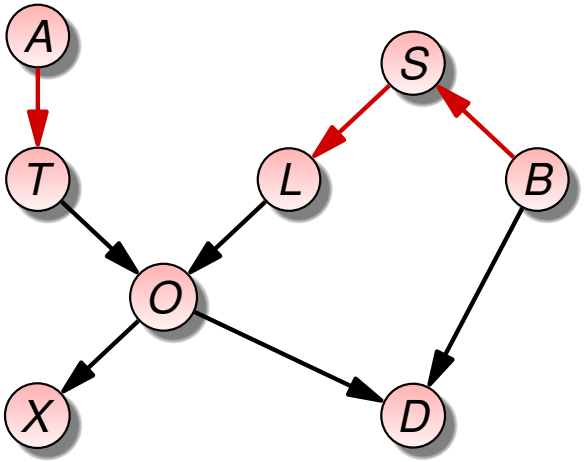
Résultat : représentant de la classe d'équivalence de Markov

- CPDAG : completed partially directed acyclic graph

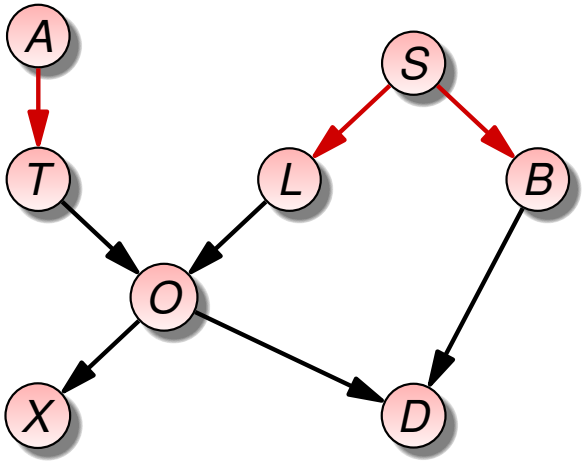
6 possibilités :



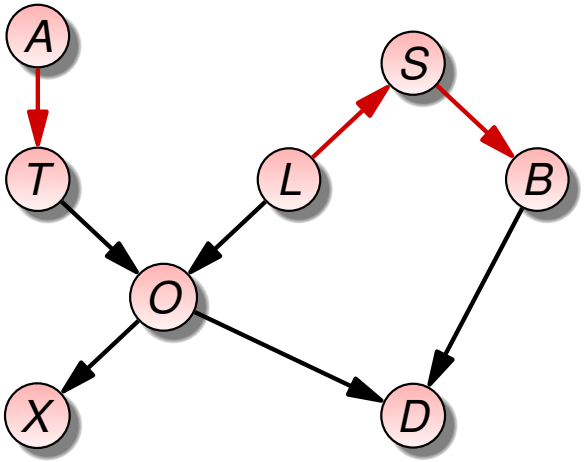
Après $R1$



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
- $N_{xy\mathbf{z}}$: nombre d'occurrences de $(X = x, Y = y, \mathbf{Z} = \mathbf{z})$ dans $\mathbf{D}$

Tests d'indépendance conditionnelle (1/2)

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

Test du χ^2

$$\chi^2_{statistics}(X, Y|\mathbf{Z}) = \sum_{x \in \Omega_X} \sum_{y \in \Omega_Y} \sum_{\mathbf{z} \in \Omega_{\mathbf{Z}}} \frac{\left(N_{xy\mathbf{z}} - \frac{N_{x\mathbf{z}} N_{y\mathbf{z}}}{N_{\mathbf{z}}}\right)^2}{\frac{N_{x\mathbf{z}} N_{y\mathbf{z}}}{N_{\mathbf{z}}}}$$

Tests d'indépendance conditionnelle (1/2)

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

Test du χ^2

$$\chi^2_{statistics}(X, Y|\mathbf{Z}) = \sum_{x \in \Omega_X} \sum_{y \in \Omega_Y} \sum_{\mathbf{z} \in \Omega_{\mathbf{Z}}} \frac{\left(N_{xy\mathbf{z}} - \frac{N_{x\mathbf{z}} N_{y\mathbf{z}}}{N_{\mathbf{z}}}\right)^2}{\frac{N_{x\mathbf{z}} N_{y\mathbf{z}}}{N_{\mathbf{z}}}}$$

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

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

Tests d'indépendance conditionnelle (2/2)

Test du G^2

$$G^2_{statistics}(X, Y|\mathbf{Z}) = 2 \sum_{x \in \Omega_X} \sum_{y \in \Omega_Y} \sum_{\mathbf{z} \in \Omega_\mathbf{Z}} N_{xy\mathbf{z}} \ln \frac{N_{xy\mathbf{z}} N_\mathbf{z}}{N_{x\mathbf{z}} N_{y\mathbf{z}}}$$

Tests d'indépendance conditionnelle (2/2)

Test du G^2

$$G^2_{statistics}(X, Y|\mathbf{Z}) = 2 \sum_{x \in \Omega_X} \sum_{y \in \Omega_Y} \sum_{\mathbf{z} \in \Omega_\mathbf{Z}} N_{xy\mathbf{z}} \ln \frac{N_{xy\mathbf{z}} N_\mathbf{z}}{N_{x\mathbf{z}} N_{y\mathbf{z}}}$$

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

- En pratique, tests du G^2 plus robustes que ceux du χ^2

Problèmes de l'algorithme PC

- Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête X − Y t.q. |Adj(X)\{Y}| ≥ d faire
2     ...
3     Supprimer l'arête X − Y de 𝒢
4     ...
```

Problèmes de l'algorithme PC

- Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête X − Y t.q. |Adj(X)\{Y}| ≥ d faire
2     ...
3     Supprimer l'arête X − Y de 𝒢
4     ...
```

$\Longrightarrow$ PC-stable, variante de Colombo et Maathuis (2014)

Problèmes de l'algorithme PC

- Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête X − Y t.q. |Adj(X)\{Y}| ≥ d faire
2     ...
3     Supprimer l'arête X − Y de 𝒢
4     ...
```

  $\Longrightarrow$ PC-stable, variante de Colombo et Maathuis (2014)

- $|\mathbf{D}|$ petite $\Longrightarrow$ tests χ^2 et G^2 peu fiables

Problèmes de l'algorithme PC

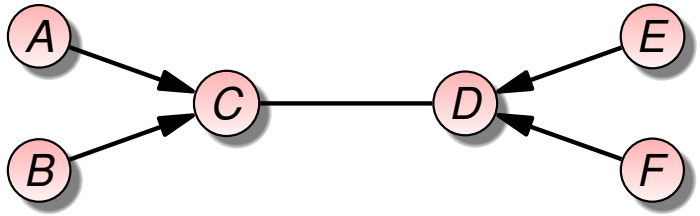
- Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête X − Y t.q. |Adj(X)\{Y}| ≥ d faire
2 |   ...
3 |   Supprimer l'arête X − Y de 𝒢
4 |   ...
```

  $\Longrightarrow$ PC-stable, variante de Colombo et Maathuis (2014)

- $|\mathbf{D}|$ petite $\Longrightarrow$ tests χ^2 et G^2 peu fiables

- Phase 3 : pas d'orientation possible :



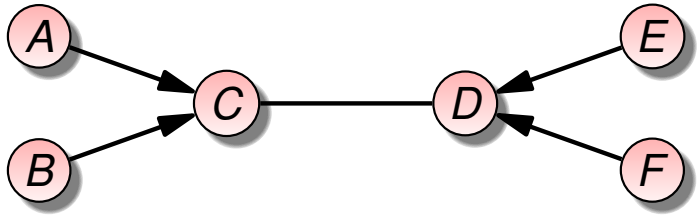
Problèmes de l'algorithme PC

- Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête X − Y t.q. |Adj(X)\{Y}| ≥ d faire
2 |   ...
3 |   Supprimer l'arête X − Y de 𝒢
4 |   ...
```

  $\Longrightarrow$ PC-stable, variante de Colombo et Maathuis (2014)

- $|\mathbf{D}|$ petite $\Longrightarrow$ 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)

Problèmes de l'algorithme PC

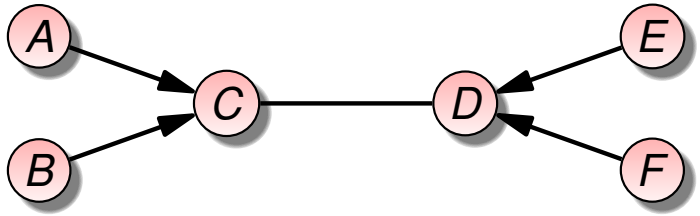
- Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête X − Y t.q. |Adj(X)\{Y}| ≥ d faire
2 |   ...
3 |   Supprimer l'arête X − Y de 𝒢
4 |   ...
```

  $\Longrightarrow$ PC-stable, variante de Colombo et Maathuis (2014)

- $|\mathbf{D}|$ petite $\Longrightarrow$ 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

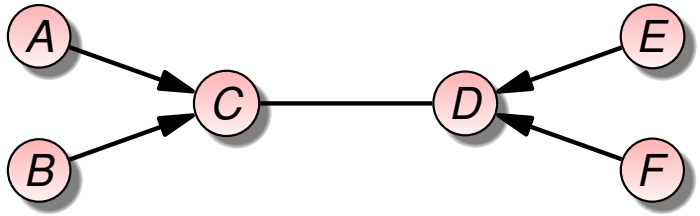
Problèmes de l'algorithme PC

- Résultat dépendant de l'ordre des calculs :

```
1 pour chaque arête X − Y t.q. |Adj(X)\{Y}| ≥ d faire
2 |   ...
3 |   Supprimer l'arête X − Y de 𝒢
4 |   ...
```

  $\Longrightarrow$ PC-stable, variante de Colombo et Maathuis (2014)

- $|\mathbf{D}|$ petite $\Longrightarrow$ 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)

    $\Longrightarrow$ IC* Verma (1993), FCI Spirtes, Glymour et Scheines (2000)

Bibliographie

- Colombo D. et Maathuis M.H. (2014) « Order-Independent Constraint-Based Causal Structure Learning », Journal of Machine Learning Research, 15 :3921–3962
- 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

Bibliographie

- 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

Bibliographie

- 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