AIGM 06/12/2018

APPRENTISSAGE ET SÉLECTION DE RÉSEAUX BAYÉSIENS DYNAMIQUES POUR LES PROCESSUS ONLINE NON-STATIONNAIRES.

M. Hourbracq[1,2] P-H. Wuillemin[1] C. Gonzales[1] P. Baumard[2]

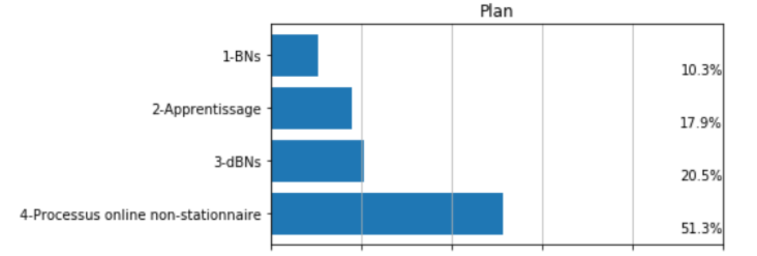
December 6, 2018

[1] Laboratoire d'Informatique de Paris VI (UPMC, CNRS UMR7606) 75004 Paris, France

[2] Akheros S.A.S. Paris, France

PLAN

PLAN


Plan
1-BNs
2-Apprentissage
3-dBNs
4-Processus online non-stationnaire
10.3%
17.9%
20.5%
51.3%

LOIS JOINTES DISCRÈTES

		A		
C	B	0	1	2
0	0	0.0143	0.0705	0.0591
1		0.0373	0.0027	0.0439
2		0.0283	0.0868	0.0570
0	1	0.0107	0.0529	0.0443
1		0.0280	0.0020	0.0329
2		0.0213	0.0651	0.0427
0	2	0.0107	0.0529	0.0443
1		0.0280	0.0020	0.0329
2		0.0213	0.0651	0.0427

$P(A, B, C)$

LOIS JOINTES DISCRÈTES

		A		
C	B	0	1	2
0		0.0143	0.0705	0.0591
1	0	0.0373	0.0027	0.0439
2		0.0283	0.0868	0.0570
0		0.0107	0.0529	0.0443
1	1	0.0280	0.0020	0.0329
2		0.0213	0.0651	0.0427
0		0.0107	0.0529	0.0443
1	2	0.0280	0.0020	0.0329
2		0.0213	0.0651	0.0427

$P(A, B, C)$

	A		
B	0	1	2
0	0.0800	0.1600	0.1600
1	0.0600	0.1200	0.1200
2	0.0600	0.1200	0.1200

$P(A, B) = \sum_C P(A, B, C)$

C	B	A = 0	A = 1	A = 2
0	0	0.0143	0.0705	0.0591
1	0	0.0373	0.0027	0.0439
2	0	0.0283	0.0868	0.0570
0	1	0.0107	0.0529	0.0443
1	1	0.0280	0.0020	0.0329
2	1	0.0213	0.0651	0.0427
0	2	0.0107	0.0529	0.0443
1	2	0.0280	0.0020	0.0329
2	2	0.0213	0.0651	0.0427

$P(A, B, C)$

B	A = 0	A = 1	A = 2
0	0.0800	0.1600	0.1600
1	0.0600	0.1200	0.1200
2	0.0600	0.1200	0.1200

$P(A, B) = \sum_C P(A, B, C)$

A = 0	A = 1	A = 2
0.2000	0.4000	0.4000

$P(A) = \sum_{B,C} P(A, B, C)$

B = 0	B = 1	B = 2
0.4000	0.3000	0.3000

$P(B) = \sum_{A,C} P(A, B, C)$

LOIS JOINTES DISCRÈTES

C	B	A = 0	A = 1	A = 2
0	0	0.0143	0.0705	0.0591
1	0	0.0373	0.0027	0.0439
2	0	0.0283	0.0868	0.0570
0	1	0.0107	0.0529	0.0443
1	1	0.0280	0.0020	0.0329
2	1	0.0213	0.0651	0.0427
0	2	0.0107	0.0529	0.0443
1	2	0.0280	0.0020	0.0329
2	2	0.0213	0.0651	0.0427

A = 0	A = 1	A = 2
0.2000	0.4000	0.4000

B = 0	B = 1	B = 2
0.4000	0.3000	0.3000

B	A	C = 0	C = 1	C = 2
0	0	0.1790	0.4668	0.3542
1	0	0.1790	0.4668	0.3542
2	0	0.1790	0.4668	0.3542
0	1	0.4405	0.0168	0.5427
1	1	0.4405	0.0168	0.5427
2	1	0.4405	0.0168	0.5427
0	2	0.3695	0.2744	0.3561
1	2	0.3695	0.2744	0.3561
2	2	0.3695	0.2744	0.3561

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

LOIS JOINTES DISCRÈTES

		A		
C	B	0	1	2
0	0	0.0143	0.0705	0.0591
1	0	0.0373	0.0027	0.0439
2	0	0.0283	0.0868	0.0570
0	1	0.0107	0.0529	0.0443
1	1	0.0280	0.0020	0.0329
2	1	0.0213	0.0651	0.0427
0	2	0.0107	0.0529	0.0443
1	2	0.0280	0.0020	0.0329
2	2	0.0213	0.0651	0.0427

A		
0	1	2
0.2000	0.4000	0.4000

B		
0	1	2
0.4000	0.3000	0.3000

		C		
B	A	0	1	2
0	0	0.1790	0.4668	0.3542
1	0	0.1790	0.4668	0.3542
2	0	0.1790	0.4668	0.3542
0	1	0.4405	0.0168	0.5427
1	1	0.4405	0.0168	0.5427
2	1	0.4405	0.0168	0.5427
0	2	0.3695	0.2744	0.3561
1	2	0.3695	0.2744	0.3561
2	2	0.3695	0.2744	0.3561

$$P(A, B, C) \quad = \quad P(A) \quad * \quad P(B) \quad * \quad P(C|B, A)$$

$$A \wedge B \mapsto C$$

LOIS JOINTES DISCRÈTES

		A		
C	B	0	1	2
0	0	0.0143	0.0705	0.0591
1	0	0.0373	0.0027	0.0439
2	0	0.0283	0.0868	0.0570
0	1	0.0107	0.0529	0.0443
1	1	0.0280	0.0020	0.0329
2	1	0.0213	0.0651	0.0427
0	2	0.0107	0.0529	0.0443
1	2	0.0280	0.0020	0.0329
2	2	0.0213	0.0651	0.0427

A		
0	1	2
0.2000	0.4000	0.4000

B		
0	1	2
0.4000	0.3000	0.3000

	C		
A	0	1	2
0	0.1790	0.4668	0.3542
1	0.4405	0.0168	0.5427
2	0.3695	0.2744	0.3561

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

C	B	A = 0	A = 1	A = 2
0	0	0.0143	0.0705	0.0591
1	0	0.0373	0.0027	0.0439
2	0	0.0283	0.0868	0.0570
0	1	0.0107	0.0529	0.0443
1	1	0.0280	0.0020	0.0329
2	1	0.0213	0.0651	0.0427
0	2	0.0107	0.0529	0.0443
1	2	0.0280	0.0020	0.0329
2	2	0.0213	0.0651	0.0427

$P(A, B, C)$ =

A = 0	A = 1	A = 2
0.2000	0.4000	0.4000

$P(A)$ ∗

B = 0	B = 1	B = 2
0.4000	0.3000	0.3000

$P(B)$ ∗

A	C = 0	C = 1	C = 2
0	0.1790	0.4668	0.3542
1	0.4405	0.0168	0.5427
2	0.3695	0.2744	0.3561

$P(C|A)$

$A \mapsto C$

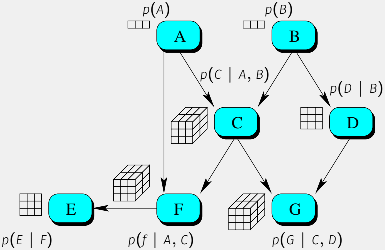
$P(A, B, C, D, E, F, G)$?

$P(A, B, C, D, E, F, G)$?

K^7 paramètres

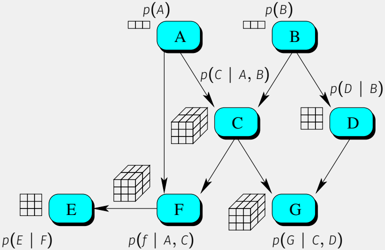
$P(A, B, C, D, E, F, G)$?

K^7 paramètres



$P(A, B, C, D, E, F, G)$?

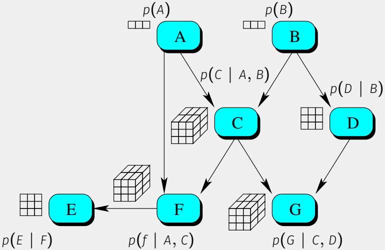
K^7 paramètres



$2K + 2K^2 + 3K^3$ paramètres

$P(A, B, C, D, E, F, G)$?

K^7 paramètres

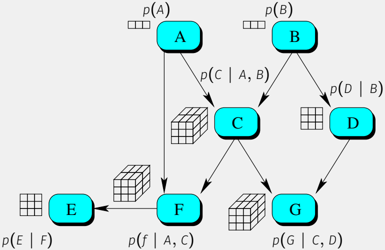


$2K + 2K^2 + 3K^3$ paramètres

Indépendances dans un BN

$P(A, B, C, D, E, F, G)$?

K^7 paramètres



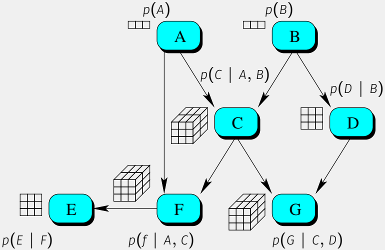
$2K + 2K^2 + 3K^3$ paramètres

Indépendances dans un BN

$X_i \perp\!\!\!\perp \text{non-desc}_{X_i} \,|\, \text{parents}_{X_i}$

$P(A, B, C, D, E, F, G)$?

K^7 paramètres



$2K + 2K^2 + 3K^3$ paramètres

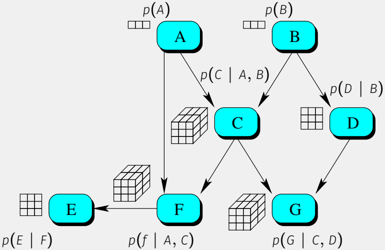
Indépendances dans un BN

$X_i \perp\!\!\!\perp \text{non-desc}_{X_i} \mid \text{parents}_{X_i}$

Factorisation dans un BN

$P(A, B, C, D, E, F, G)$?

K^7 paramètres



$2K + 2K^2 + 3K^3$ paramètres

Indépendances dans un BN

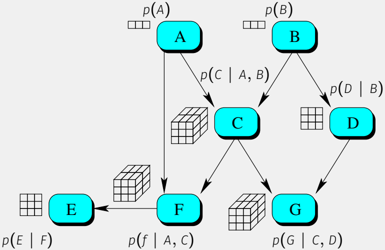
$X_i \perp\!\!\!\perp \text{non-desc}_{X_i} \mid \text{parents}_{X_i}$

Factorisation dans un BN

$P(X_1, \cdots, X_n) = \prod_i P(X_i \mid \text{parents}_{X_i})$

$P(A, B, C, D, E, F, G)$?

K^7 paramètres



$2K + 2K^2 + 3K^3$ paramètres

Indépendances dans un BN

$X_i \perp\!\!\!\perp \text{non-desc}_{X_i} \mid \text{parents}_{X_i}$

Factorisation dans un BN

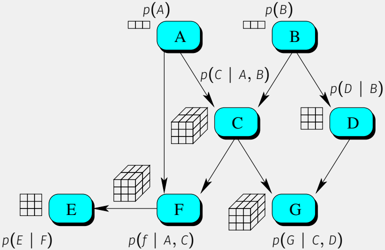
$P(X_1, \cdots, X_n) = \prod_i P(X_i \mid \text{parents}_{X_i})$

Inférences dans un BN

RÉSEAUX BAYÉSIENS (DISCRETS) DANS UNE COQUILLE DE NOIX

$P(A, B, C, D, E, F, G)$?

K^7 paramètres



$2K + 2K^2 + 3K^3$ paramètres

Indépendances dans un BN

$X_i \perp\!\!\!\perp \text{non-desc}_{X_i} \mid \text{parents}_{X_i}$

Factorisation dans un BN

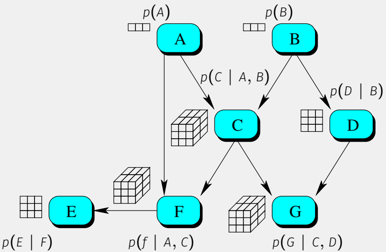
$P(X_1, \cdots, X_n) = \prod_i P(X_i \mid \text{parents}_{X_i})$

Inférences dans un BN

- $P(A|G = 1)$

$P(A, B, C, D, E, F, G)$?

K^7 paramètres



$2K + 2K^2 + 3K^3$ paramètres

Indépendances dans un BN

$X_i \perp\!\!\!\perp \text{non-desc}_{X_i} \mid \text{parents}_{X_i}$

Factorisation dans un BN

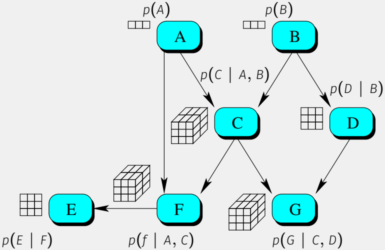
$P(X_1, \cdots, X_n) = \prod_i P(X_i \mid \text{parents}_{X_i})$

Inférences dans un BN

- $P(A|G = 1)$
- $P(G|A = 0)$

$P(A, B, C, D, E, F, G)$?

K^7 paramètres



$2K + 2K^2 + 3K^3$ paramètres

Indépendances dans un BN

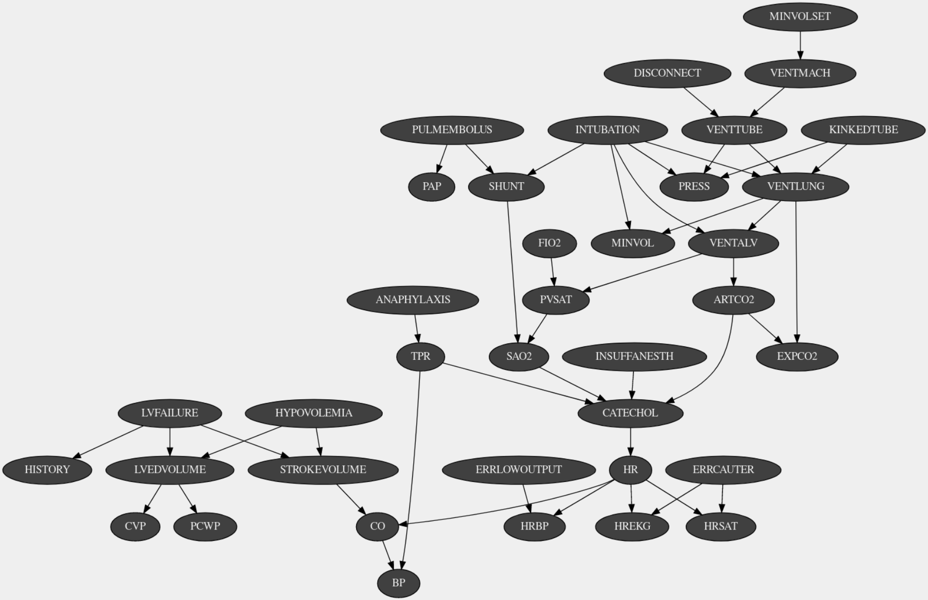
$X_i \perp\!\!\!\perp \text{non-desc}_{X_i} \mid \text{parents}_{X_i}$

Factorisation dans un BN

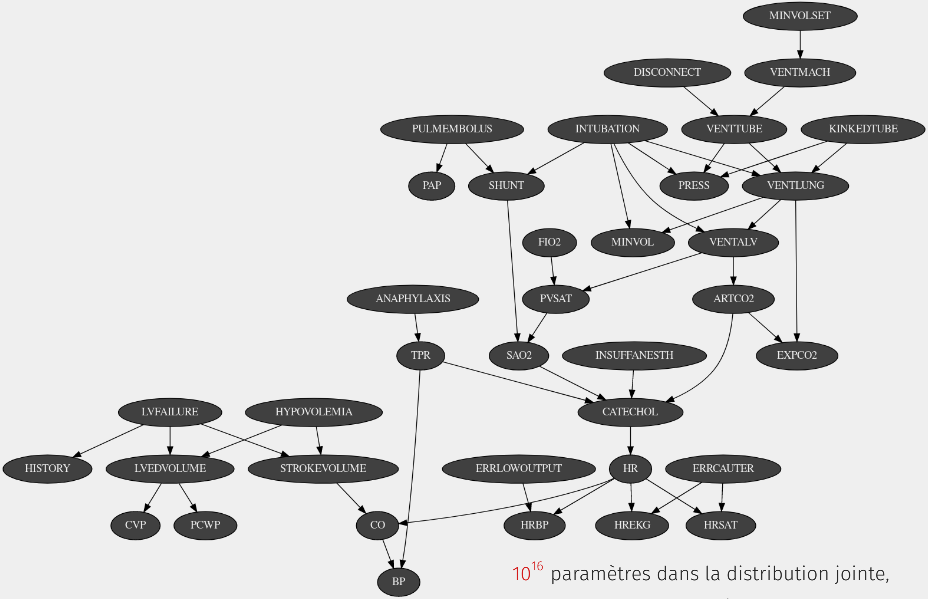
$P(X_1, \cdots, X_n) = \prod_i P(X_i \mid \text{parents}_{X_i})$

Inférences dans un BN

- $P(A|G = 1)$
- $P(G|A = 0)$
- $P(E \mid A = 0, G = 1)$


MINVOLSET
DISCONNECT
VENTMACH
PULMEMBOLUS
INTUBATION
VENTTUBE
KINKEDTUBE
PAP
SHUNT
PRESS
VENTLUNG
FIO2
MINVOL
VENTALV
ANAPHYLAXIS
PVSAT
ARTCO2
TPR
SAO2
INSUFFANESTH
EXPCO2
LVFAILURE
HYPOVOLEMIA
CATECHOL
HISTORY
LVEDVOLUME
STROKEVOLUME
ERRLOWOUTPUT
HR
ERRCAUTER
CVP
PCWP
CO
HRBP
HREKG
HRSAT
BP

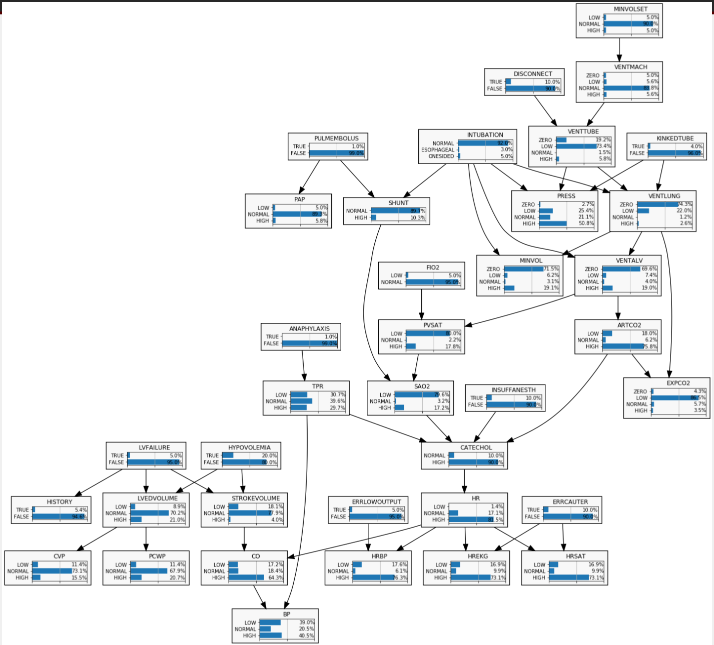
INFÉRENCES DANS UN BN



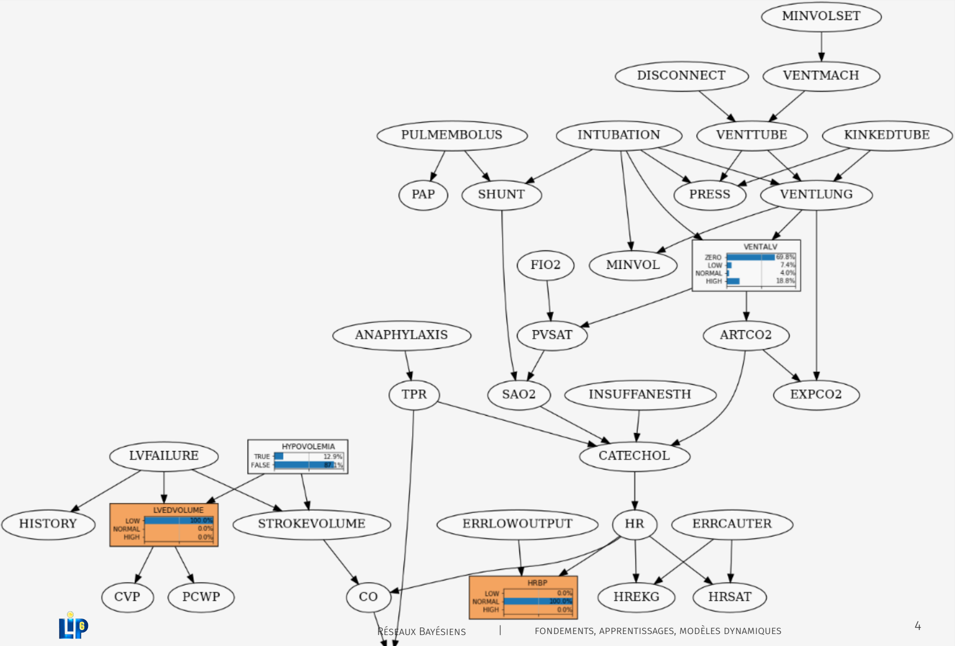
10^{16} paramètres dans la distribution jointe,

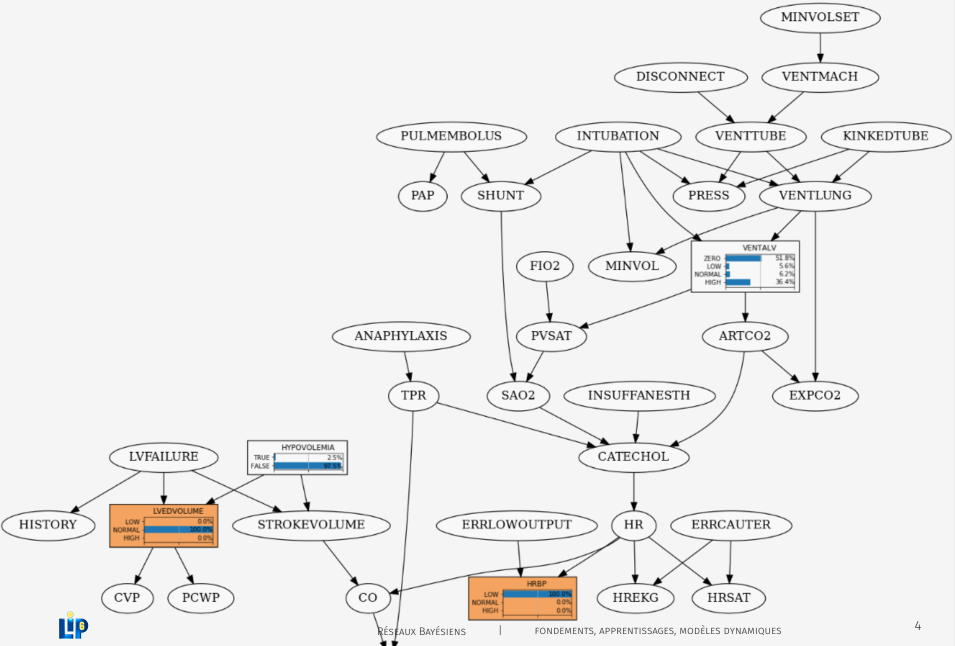
752 paramètres dans le BN !

INFÉRENCES DANS UN BN



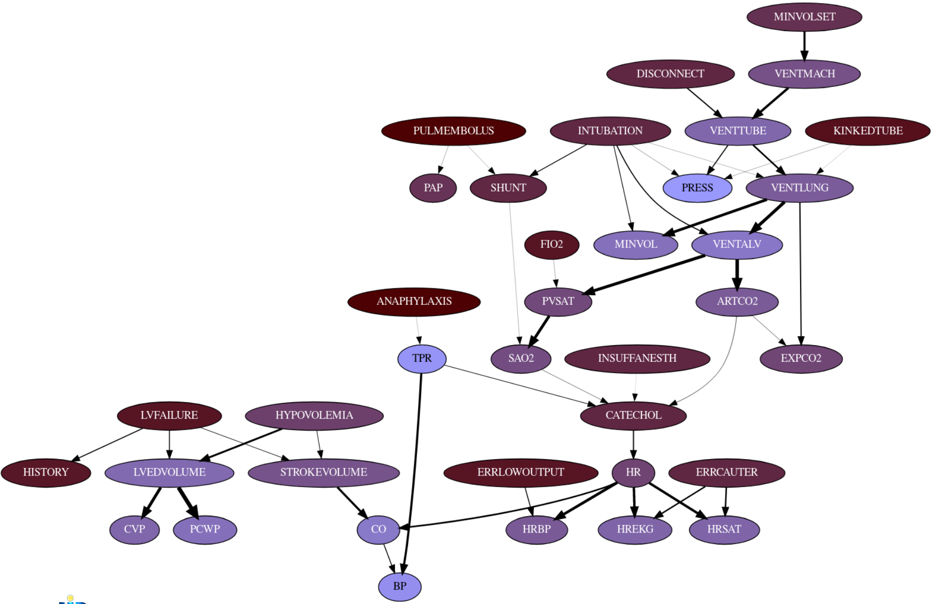
INFÉRENCES DANS UN BN




MINVOLSET
DISCONNECT
VENTMACH
PULMEMBOLUS
INTUBATION
VENTTUBE
KINKEDTUBE
PAP
SHUNT
PRESS
VENTLUNG
FIO2
MINVOL
VENTALV
ZERO 51.8%
LOW 5.6%
NORMAL 6.2%
HIGH 36.4%
ANAPHYLAXIS
PVSAT
ARTCO2
TPR
SAO2
INSUFFANESTH
EXPCO2
LVFAILURE
HYPOVOLEMIA
TRUE 2.5%
FALSE 97.5%
CATECHOL
HISTORY
LVEDVOLUME
LOW 0.0%
NORMAL 100.0%
HIGH 0.0%
STROKEVOLUME
ERRLOWOUTPUT
HR
ERRCAUTER
CVP
PCWP
CO
HRBP
LOW 100.0%
NORMAL 0.0%
HIGH 0.0%
HREKG
HRSAT

AUTRES CALCULS DANS UN BN

AUTRES CALCULS DANS UN BN



APPRENDRE QUOI ?

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer,

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données

APPRENDRE QUOI ?

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données et de connaissances a priori :

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données et de connaissances a priori :

- La structure du réseau bayésien (X parent de Y ?)

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données et de connaissances a priori :

- La structure du réseau bayésien (X parent de Y ?)
- Les paramètres du réseau bayésien ($P\big(X = 0 \mid Y = 1\big)$?)

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données et de connaissances a priori :

- La structure du réseau bayésien (X parent de Y ?)
- Les paramètres du réseau bayésien ($P\big(X = 0 \mid Y = 1\big)$?)

La base de données peut être :

APPRENDRE QUOI ?

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données et de connaissances a priori :

- La structure du réseau bayésien (X parent de Y ?)
- Les paramètres du réseau bayésien ($P\big(X = 0 \mid Y = 1\big)$?)

La base de données peut être :

- complète,
- incomplète.

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données et de connaissances a priori :

- La structure du réseau bayésien (X parent de Y ?)
- Les paramètres du réseau bayésien ($P\left(X = 0 \mid Y = 1\right)$?)

La base de données peut être :

- complète,
- incomplète.

Ce qui donne 4 cadres principaux de l'apprentissage dans les réseaux Bayésiens :
"Apprentissage de {paramètres |structure} avec données {complètes |incomplètes}".

Les connaissances a priori sont très variables;

Apprentissage dans les réseaux bayésiens

L'apprentissage a pour but d'estimer, à partir d'une base de données et de connaissances a priori :

- La structure du réseau bayésien (X parent de Y ?)
- Les paramètres du réseau bayésien ($P\big(X = 0 \mid Y = 1\big)$?)

La base de données peut être :

- complète,
- incomplète.

Ce qui donne 4 cadres principaux de l'apprentissage dans les réseaux Bayésiens :
"Apprentissage de {paramètres |structure} avec données {complètes |incomplètes}".

Les connaissances a priori sont très variables; par exemple :

- sous-structure du BN connue,
- Loi a priori pour certaines variables, etc.

- But :

APPRENTISSAGE DE LA STRUCTURE, DONNÉES COMPLÈTES

- **But** : obtenir automatiquement une structure de réseau bayésien à partir de données.

APPRENTISSAGE DE LA STRUCTURE, DONNÉES COMPLÈTES

- **But** : obtenir automatiquement une structure de réseau bayésien à partir de données.
- **En théorie** : Test du χ^2 plus énumération de tous les modèles possibles : OK

APPRENTISSAGE DE LA STRUCTURE, DONNÉES COMPLÈTES

- **But** : obtenir automatiquement une structure de réseau bayésien à partir de données.
- **En théorie** : Test du χ^2 plus énumération de tous les modèles possibles : OK
- **En pratique** : Beaucoup de problème mais avant tout :

- **But** : obtenir automatiquement une structure de réseau bayésien à partir de données.
- **En théorie** : Test du χ^2 plus énumération de tous les modèles possibles : OK
- **En pratique** : Beaucoup de problème mais avant tout :

Espace des réseaux bayésiens (Robinson, 1977)

Le nombre de structures possibles pour n nœuds est super-exponentiel.

$$NS(n) = \begin{cases} 1 & , n \leq 1 \\ \sum_{i=1}^{n} (-1)^{i+1} \cdot C_i^n \cdot 2^{i \cdot (n-i)} \cdot NS(n-1) & , n > 1 \end{cases}$$

Robinson (1977) *Counting unlabelled acyclid digraphs*. In Lecture Notes in Mathematics : Combinatorial Mathematics V

La recherche exhaustive n'est pas possible. L'espace est bien trop grand :
$NS(10) \approx 4.2 \cdot 10^{18}$!

Tableau général de l'apprentissage de structure

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)
  - Dans l'espace des structures

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)
  - Dans l'espace des structures (BN ou équivalent de Markov),

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)
  - Dans l'espace des structures (BN ou équivalent de Markov),
  - algorithmes essayant de maximiser un score

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)
  - Dans l'espace des structures (BN ou équivalent de Markov),
  - algorithmes essayant de maximiser un score (entropie, AIC, BIC, MDL,BD, BDe, BDeu, $\cdots$).

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)
  - Dans l'espace des structures (BN ou équivalent de Markov),
  - algorithmes essayant de maximiser un score (entropie, AIC, BIC, MDL,BD, BDe, BDeu, $\cdots$).

Classe d'équivalence de Markov

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)
  - Dans l'espace des structures (BN ou équivalent de Markov),
  - algorithmes essayant de maximiser un score (entropie, AIC, BIC, MDL,BD, BDe, BDeu, $\cdots$).

Classe d'équivalence de Markov

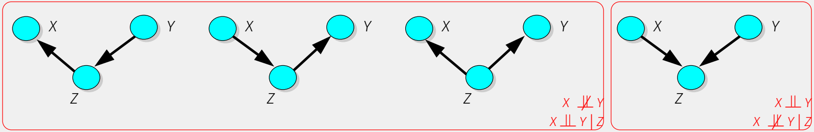
Deux réseaux bayésiens sont équivalents si ils représentent le même modèle d'indépendance.

Tableau général de l'apprentissage de structure

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
  - Autres tests d'indépendances Entropie : 3off2, miic
- Recherche heuristique (score)
  - Dans l'espace des structures (BN ou équivalent de Markov),
  - algorithmes essayant de maximiser un score (entropie, AIC, BIC, MDL,BD, BDe, BDeu, $\cdots$).

Classe d'équivalence de Markov

Deux réseaux bayésiens sont équivalents si ils représentent le même modèle d'indépendance.



RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.

RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.
   - Retirer des arêtes à partir du graphe complet.

RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.
   - Retirer des arêtes à partir du graphe complet.
2. Détecter les V-structures et les orientations qu'elles impliquent.

RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.
   - Retirer des arêtes à partir du graphe complet.
2. Détecter les V-structures et les orientations qu'elles impliquent.
3. Finaliser les orientations en restant dans la même classe d'équivalence de Markov.

RECHERCHE DE RELATION SYMÉTRIQUE

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.
   - Retirer des arêtes à partir du graphe complet.
2. Détecter les V-structures et les orientations qu'elles impliquent.
3. Finaliser les orientations en restant dans la même classe d'équivalence de Markov.

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.
   - Retirer des arêtes à partir du graphe complet.
2. Détecter les V-structures et les orientations qu'elles impliquent.
3. Finaliser les orientations en restant dans la même classe d'équivalence de Markov.

Écueils principaux

En terme statistique, les relations testables sont symétriques : corrélation ou indépendance entre variables aléatoires, information mutuelle, etc..

Par contre, une fois des relations 2 à 2 trouvées, il s'agit de tester certaines indépendances conditionnelles (V-structure) qui forcent les orientations.

Principe de base (IC, IC*, PC, FCI)

1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.
   - Retirer des arêtes à partir du graphe complet.
2. Détecter les V-structures et les orientations qu'elles impliquent.
3. Finaliser les orientations en restant dans la même classe d'équivalence de Markov.

Écueils principaux

Très grand nombre de tests d'indépendances, chaque test étant très sensible au nombre de données disponibles.

ALGORITHMES DIRIGÉS PAR UNE HEURISTIQUE

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.).

ALGORITHMES DIRIGÉS PAR UNE HEURISTIQUE

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** :

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données ($\max L(T, \Theta : D)$).

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données ($\max L(T, \Theta : D)$).
2. **Rasoir d'Occam** :

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données ($\max L(T, \Theta : D)$).
2. **Rasoir d'Occam** : Privilégier les topologies T simples aux topologies complexes ($\min Dim(T)$).

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données (max $L(T, \Theta : D)$).
2. **Rasoir d'Occam** : Privilégier les topologies T simples aux topologies complexes (min $Dim(T)$).
3. **Consistance locale** :

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données ($\max L(T, \Theta : D)$).
2. **Rasoir d'Occam** : Privilégier les topologies T simples aux topologies complexes ($\min Dim(T)$).
3. **Consistance locale** : Ajouter un arc 'utile' devrait augmenter le score. Ajouter un arc 'inutile' devrait diminuer le score.

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données ($\max L(T, \Theta : D)$).
2. **Rasoir d'Occam** : Privilégier les topologies T simples aux topologies complexes ($\min Dim(T)$).
3. **Consistance locale** : Ajouter un arc 'utile' devrait augmenter le score. Ajouter un arc 'inutile' devrait diminuer le score.
4. **Score équivalence** :

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données ($\max L(T, \Theta : D)$).
2. **Rasoir d'Occam** : Privilégier les topologies T simples aux topologies complexes ($\min Dim(T)$).
3. **Consistance locale** : Ajouter un arc 'utile' devrait augmenter le score. Ajouter un arc 'inutile' devrait diminuer le score.
4. **Score équivalence** : Deux réseaux bayésiens Markov-équivalents devraient avoir le même score.

ALGORITHMES DIRIGÉS PAR UNE HEURISTIQUE

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données ($\max L(T, \Theta : D)$).
2. **Rasoir d'Occam** : Privilégier les topologies T simples aux topologies complexes ($\min Dim(T)$).
3. **Consistance locale** : Ajouter un arc 'utile' devrait augmenter le score. Ajouter un arc 'inutile' devrait diminuer le score.
4. **Score équivalence** : Deux réseaux bayésiens Markov-équivalents devraient avoir le même score.
5. **Décomposition locale** :

La recherche exhaustive des relations d'indépendances est inatteignable (nombre de tests prohibitifs, quantité de données nécessaires trop importantes, etc.). Donc utilisation d'une heuristique permettant de quantifier l'adéquation d'une structure à une base de données.

Propriétés des scores

Soient D la base de donnée, T la topologie du réseau bayésien candidat et Θ ses paramètres.Pour qu'un score (une fonction calculée sur un réseau bayésien) soit considéré comme une bonne heuristique, on peut lui demander :

1. **Vraisemblance** : Coller le mieux aux données (max $L(T, \Theta : D)$).
2. **Rasoir d'Occam** : Privilégier les topologies T simples aux topologies complexes (min $Dim(T)$).
3. **Consistance locale** : Ajouter un arc 'utile' devrait augmenter le score. Ajouter un arc 'inutile' devrait diminuer le score.
4. **Score équivalence** : Deux réseaux bayésiens Markov-équivalents devraient avoir le même score.
5. **Décomposition locale** : Calculer la modification du score par l'ajout/retrait d'un arc ne doit pas imposer de re-calculer tout le score mais seulement une partie, locale à l'arc modifié.

RECHERCHE LOCALE À BASE DE SCORES

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,

RECHERCHE LOCALE À BASE DE SCORES

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires

RECHERCHE LOCALE À BASE DE SCORES

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires
- Soit un score (heurisitique) calculable localement.

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires
- Soit un score (heurisitique) calculable localement.

La recherche locale est alors une séquence de voisins telle qu'à partir du point initial, tout élément ultérieur de la séquence augmente le score. (*Greedy Search*).

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires
- Soit un score (heurisitique) calculable localement.

La recherche locale est alors une séquence de voisins telle qu'à partir du point initial, tout élément ultérieur de la séquence augmente le score. (*Greedy Search*).

Recherche locale dans les réseaux bayésiens

- L'espace est l'espace des réseaux bayésiens (énorme)

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires
- Soit un score (heurisitique) calculable localement.

La recherche locale est alors une séquence de voisins telle qu'à partir du point initial, tout élément ultérieur de la séquence augmente le score. (*Greedy Search*).

Recherche locale dans les réseaux bayésiens

- L'espace est l'espace des réseaux bayésiens (énorme)
- Soit une structure initiale

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires
- Soit un score (heurisitique) calculable localement.

La recherche locale est alors une séquence de voisins telle qu'à partir du point initial, tout élément ultérieur de la séquence augmente le score. (*Greedy Search*).

Recherche locale dans les réseaux bayésiens

- L'espace est l'espace des réseaux bayésiens (énorme)
- Soit une structure initiale
- Les opérations de base

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires
- Soit un score (heurisitique) calculable localement.

La recherche locale est alors une séquence de voisins telle qu'à partir du point initial, tout élément ultérieur de la séquence augmente le score. (*Greedy Search*).

Recherche locale dans les réseaux bayésiens

- L'espace est l'espace des réseaux bayésiens (énorme)
- Soit une structure initiale
- Les opérations de base : ajout/suppression/modification d'un arc (dans le domaine de validité)

RECHERCHE LOCALE À BASE DE SCORES

L'algorithme de recherche locale est un algorithme générique qui ne demande que quelques hypothèses de base :

Recherche locale

- Soit un espace de recherche,
- Soit une notion de voisinage définie par des opérations élémentaires
- Soit un score (heurisitique) calculable localement.

La recherche locale est alors une séquence de voisins telle qu'à partir du point initial, tout élément ultérieur de la séquence augmente le score. (*Greedy Search*).

Recherche locale dans les réseaux bayésiens

- L'espace est l'espace des réseaux bayésiens (énorme)
- Soit une structure initiale
- Les opérations de base : ajout/suppression/modification d'un arc (dans le domaine de validité)

QUELQUES SCORES : AIC/BIC/MDL

Idée de base : Il faut maximiser la vraisemblance tout en minimisant la dimension.

Idée de base : Il faut maximiser la vraisemblance tout en minimisant la dimension.

score AIC (Akaike, 70)

- Akaike Information Criterion

$$\mathrm{Score}_{\mathrm{AIC}}(T, D) = log_2 L(\Theta^{\mathrm{MV}}, T : D) - Dim(T)$$

Idée de base : Il faut maximiser la vraisemblance tout en minimisant la dimension.

score AIC (Akaike, 70)

- Akaike Information Criterion

$$\mathrm{Score}_{\mathrm{AIC}}(T, D) = log_2 L(\Theta^{\mathrm{MV}}, T : D) - Dim(T)$$

score BIC (Schwartz, 78)

- Bayesian Information Criterion

$$\mathrm{Score}_{\mathrm{BIC}}(T, D) = log_2 L(\Theta^{\mathrm{MV}}, T : D) - \frac{1}{2} \cdot Dim(T) \cdot log_2 N$$

QUELQUES SCORES : AIC/BIC/MDL

Idée de base : Il faut maximiser la vraisemblance tout en minimisant la dimension.

score AIC (Akaike, 70)

- Akaike Information Criterion

$$\text{Score}_{\text{AIC}}(T, D) = log_2 L(\Theta^{\text{MV}}, T : D) - Dim(T)$$

score BIC (Schwartz, 78)

- Bayesian Information Criterion

$$\text{Score}_{\text{BIC}}(T, D) = log_2 L(\Theta^{\text{MV}}, T : D) - \frac{1}{2} \cdot Dim(T) \cdot log_2 N$$

score MDL (Lam and Bacchus, 93)

- Minimum Description Length

$$\text{Score}_{\text{MDL}}(T, D) = log_2 L(\Theta^{\text{MV}}, T : D) - |\text{arcs}_T| \cdot log_2 N - c \cdot Dim(T)$$

où arcs_T est l'ensemble des arcs du graphe, c est le nombre de bits nécessaire à la représentation d'un paramètre.

dBN (dynamic BN)

dBN (dynamic BN)

Un réseau bayésien dynamique est un réseau bayésien dont les variables sont indicées par le temps t et par i : $X^{(t)} = X_1^{(t)}, \cdots, X_N^{(t)}$ et dont la distribution vérifie certaines propriétés :

dBN (dynamic BN)

Un réseau bayésien dynamique est un réseau bayésien dont les variables sont indicées par le temps t et par i : $X^{(t)} = X_1^{(t)}, \cdots, X_N^{(t)}$ et dont la distribution vérifie certaines propriétés :

- Markov ordre 1 :
  $P\big(X^{(t)} \mid X^{(0)}, \cdots, X^{(t-1)}\big) = P\big(X^{(t)} \mid X^{(t-1)}\big)$,

dBN (dynamic BN)

Un réseau bayésien dynamique est un réseau bayésien dont les variables sont indicées par le temps t et par i : $X^{(t)} = X_1^{(t)}, \cdots, X_N^{(t)}$ et dont la distribution vérifie certaines propriétés :

- Markov ordre 1 :
  $P(X^{(t)} \mid X^{(0)}, \cdots, X^{(t-1)}) = P(X^{(t)} \mid X^{(t-1)})$,
- Homogénéité :
  $P(X^{(t)} \mid X^{(t-1)}) = \cdots = P(X^{(1)} \mid X^{(0)})$.

dBN (dynamic BN)

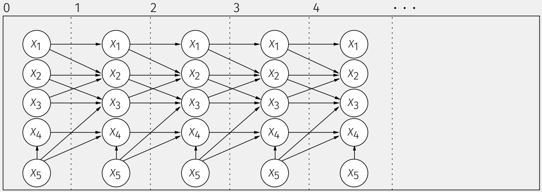
Un réseau bayésien dynamique est un réseau bayésien dont les variables sont indicées par le temps t et par i : $X^{(t)} = X_1^{(t)}, \cdots, X_N^{(t)}$ et dont la distribution vérifie certaines propriétés :

- Markov ordre 1 :
  $P\left(X^{(t)} \mid X^{(0)}, \cdots, X^{(t-1)}\right) = P\left(X^{(t)} \mid X^{(t-1)}\right),$
- Homogénéité :
  $P\left(X^{(t)} \mid X^{(t-1)}\right) = \cdots = P\left(X^{(1)} \mid X^{(0)}\right).$

dBN (dynamic BN)

Un réseau bayésien dynamique est un réseau bayésien dont les variables sont indicées par le temps t et par i : $X^{(t)} = X_1^{(t)}, \cdots, X_N^{(t)}$ et dont la distribution vérifie certaines propriétés :

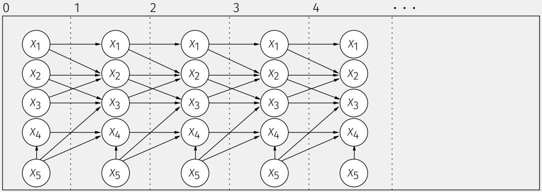
- Markov ordre 1 :
  $P(X^{(t)} \mid X^{(0)}, \cdots, X^{(t-1)}) = P(X^{(t)} \mid X^{(t-1)})$,
- Homogénéité :
  $P(X^{(t)} \mid X^{(t-1)}) = \cdots = P(X^{(1)} \mid X^{(0)})$.



dBN (dynamic BN)

Un réseau bayésien dynamique est un réseau bayésien dont les variables sont indicées par le temps t et par i : $X^{(t)} = X_1^{(t)}, \cdots, X_N^{(t)}$ et dont la distribution vérifie certaines propriétés :

- Markov ordre 1 :
  $P(X^{(t)} \mid X^{(0)}, \cdots, X^{(t-1)}) = P(X^{(t)} \mid X^{(t-1)})$,
- Homogénéité :
  $P(X^{(t)} \mid X^{(t-1)}) = \cdots = P(X^{(1)} \mid X^{(0)})$.

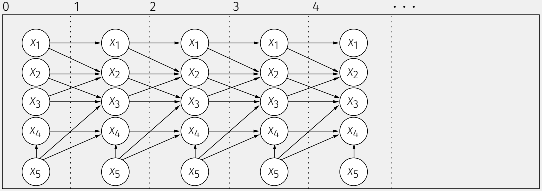


● L'adjectif *dynamique* n'est pas forcément bien choisi (puisqu'il y a homogénéité). *Temporel* ou *séquentiel* eût été de meilleur goût.

dBN (dynamic BN)

Un réseau bayésien dynamique est un réseau bayésien dont les variables sont indicées par le temps t et par i : $X^{(t)} = X_1^{(t)}, \cdots, X_N^{(t)}$ et dont la distribution vérifie certaines propriétés :

- Markov ordre 1 :
  $P(X^{(t)} \mid X^{(0)}, \cdots, X^{(t-1)}) = P(X^{(t)} \mid X^{(t-1)})$,
- Homogénéité :
  $P(X^{(t)} \mid X^{(t-1)}) = \cdots = P(X^{(1)} \mid X^{(0)})$.



- L'adjectif *dynamique* n'est pas forcément bien choisi (puisqu'il y a homogénéité). *Temporel* ou *séquentiel* eût été de meilleur goût.
- On remarque que d'après la définition, les arcs d'un dBN vont de $X^{(t-1)}$ à X^t ou restent dans le même X^t (le même *timeslice*).

2-TBN

Un réseau bayésien dynamique est défini :

LES RÉSEAUX BAYÉSIENS DYNAMIQUES (2-TBN)

2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P(X^{(0)})$

2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P(X^{(0)})$
- par les relations entre des variables à l'instant $t-1$ et ces même variables à l'instant t (*timeslice*).

2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P\big(X^{(0)}$)
- par les relations entre des variables à l'instant $t - 1$ et ces même variables à l'instant t (*timeslice*).

Cette représentation, appelée 2TBN (2 timeslice BN) permet de modéliser un BN virtuellemet infini qui en est le développement dans le temps, à partir d'un instant 0.

2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P(X^{(0)})$
- par les relations entre des variables à l'instant $t - 1$ et ces même variables à l'instant t (*timeslice*).

Cette représentation, appelée 2TBN (2 timeslice BN) permet de modéliser un BN virtuellemet infini qui en est le développement dans le temps, à partir d'un instant 0.

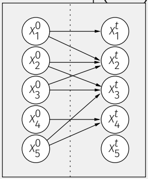
2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P(X^{(0)})$
- par les relations entre des variables à l'instant $t - 1$ et ces même variables à l'instant t (*timeslice*).

Cette représentation, appelée 2TBN (2 timeslice BN) permet de modéliser un BN virtuellemet infini qui en est le développement dans le temps, à partir d'un instant 0.

$t = 0$ $\quad$ $t \mid (t-1)$



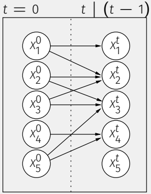
$$P(x_1^{(t)}, \ldots, x_5^{(t)} \mid x_1^{(t-1)}, \ldots, x_5^{(t-1)})$$

2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P(X^{(0)})$
- par les relations entre des variables à l'instant $t-1$ et ces même variables à l'instant t (*timeslice*).

Cette représentation, appelée 2TBN (2 timeslice BN) permet de modéliser un BN virtuellemet infini qui en est le développement dans le temps, à partir d'un instant 0.



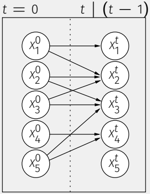
$$P(x_1^{(t)}, \dots, x_5^{(t)} \mid x_1^{(t-1)}, \dots, x_5^{(t-1)}) \qquad = P(x_1^{(t)} \mid x^{(t-1)})$$

2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P(X^{(0)})$
- par les relations entre des variables à l'instant $t-1$ et ces même variables à l'instant t (*timeslice*).

Cette représentation, appelée 2TBN (2 timeslice BN) permet de modéliser un BN virtuellemet infini qui en est le développement dans le temps, à partir d'un instant 0.



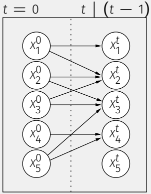
$$P(x_1^{(t)}, \ldots, x_5^{(t)} \mid x_1^{(t-1)}, \ldots, x_5^{(t-1)}) = P(x_1^{(t)} \mid x^{(t-1)}) P(x_2^{(t)} \mid x_1^{(t-1)}, x_2^{(t-1)}, x_3^{(t-1)})$$

2-TBN

Un réseau bayésien dynamique est défini :

- par les conditions initiales ($P(X^{(0)})$
- par les relations entre des variables à l'instant $t-1$ et ces même variables à l'instant t (*timeslice*).

Cette représentation, appelée 2TBN (2 timeslice BN) permet de modéliser un BN virtuellemet infini qui en est le développement dans le temps, à partir d'un instant 0.



$$P(x_1^{(t)},\ldots,x_5^{(t)} \mid x_1^{(t-1)},\ldots,x_5^{(t-1)}) = P(x_1^{(t)} \mid x^{(t-1)})\, P(x_2^{(t)} \mid x_1^{(t-1)}, x_2^{(t-1)}, x_3^{(t-1)})\, P(x_3^{(t)} \mid x_2^{(t-1)}, x_3^{(t-1)}, x_4^{(t-1)})\, P(x_4^{(t)} \mid x_4^{(t-1)}, x_5^{(t-1)})$$

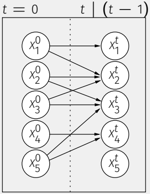
1024 contre 4+16+16+8+2=46 !!

2-TBN

Un réseau bayésien dynamique est défini :

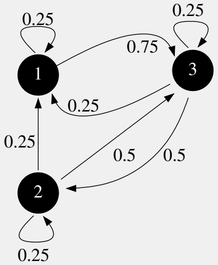
- par les conditions initiales ($P(X^{(0)})$)
- par les relations entre des variables à l'instant $t-1$ et ces même variables à l'instant t (*timeslice*).

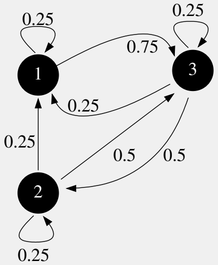
Cette représentation, appelée 2TBN (2 timeslice BN) permet de modéliser un BN virtuellemet infini qui en est le développement dans le temps, à partir d'un instant 0.



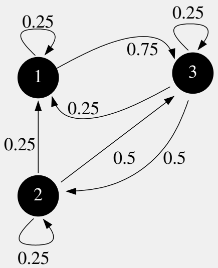
$$
\begin{aligned}
P(x_1^{(t)},\dots,x_5^{(t)} \mid x_1^{(t-1)},\dots,x_5^{(t-1)}) \quad = {} & P(x_1^{(t)} \mid x^{(t-1)}) \\
& P(x_2^{(t)} \mid x_1^{(t-1)}, x_2^{(t-1)}, x_3^{(t-1)}) \\
& P(x_3^{(t)} \mid x_2^{(t-1)}, x_3^{(t-1)}, x_4^{(t-1)}) \\
& P(x_4^{(t)} \mid x_4^{(t-1)}, x_5^{(t-1)}) \\
& P(x_5^{(t)})
\end{aligned}
$$

1024 contre 4+16+16+8+2=46 !!


0.25
0.25
0.75
1
3
0.25
0.25
0.5
0.5
2
0.25



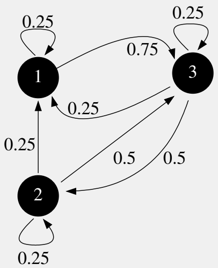
$$P(X^n \mid X^{n-1}) = \begin{pmatrix} 0.25 & 0 & 0.75 \\ 0.25 & 0.25 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{pmatrix}$$



$$P(X^n \mid X^{n-1}) = \begin{pmatrix} 0.25 & 0 & 0.75 \\ 0.25 & 0.25 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{pmatrix}$$

Chaîne de Markov

- Une variable d'état discrète (X^n) (à l'instant n).
- Paramètres du modèle :
  - Condition initiale : $P(X^O)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$



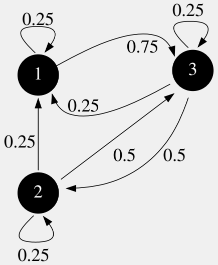
$$P(X^n \mid X^{n-1}) = \begin{pmatrix} 0.25 & 0 & 0.75 \\ 0.25 & 0.25 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{pmatrix}$$

Chaîne de Markov

- Une variable d'état discrète (X^n) (à l'instant n).
- Paramètres du modèle :
  - Condition initiale : $P(X^0)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$

Réseau bayésien dynamique équivalent :

CHAÎNE DE MARKOV ET RÉSEAUX BAYÉSIENS DYNAMIQUES

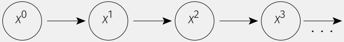


$$P(X^n \mid X^{n-1}) = \begin{pmatrix} 0.25 & 0 & 0.75 \\ 0.25 & 0.25 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{pmatrix}$$

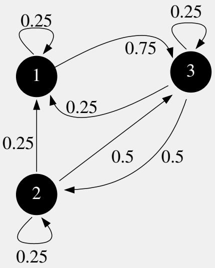
Chaîne de Markov

- Une variable d'état discrète (X^n) (à l'instant n).
- Paramètres du modèle :
  - Condition initiale : $P(X^O)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$

Réseau bayésien dynamique équivalent :

dBN : 

CHAÎNE DE MARKOV ET RÉSEAUX BAYÉSIENS DYNAMIQUES

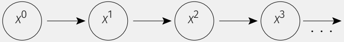


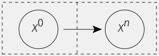
$$P(X^n \mid X^{n-1}) = \begin{pmatrix} 0.25 & 0 & 0.75 \\ 0.25 & 0.25 & 0.5 \\ 0.25 & 0.5 & 0.25 \end{pmatrix}$$

Chaîne de Markov

- Une variable d'état discrète (X^n) (à l'instant n).
- Paramètres du modèle :
  - Condition initiale : $P(X^0)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$

Réseau bayésien dynamique équivalent :

dBN : 

2TBN : 

HMM simple

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)
- Paramètres du modèle :

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)
- Paramètres du modèle :
  - Condition initiale : $P(X^O)$

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)
- Paramètres du modèle :
  - Condition initiale : $P(X^O)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)
- Paramètres du modèle :
  - Condition initiale : $P(X^0)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$
  - Modèle d'observation : $P(Y^n \mid X^n)$

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)
- Paramètres du modèle :
  - Condition initiale : $P(X^O)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$
  - Modèle d'observation : $P(Y^n \mid X^n)$

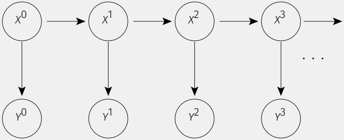
Ce qui donne, modélisé comme un réseau bayésien dynamique :

HMM simple

- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)
- Paramètres du modèle :
  - Condition initiale : $P(X^O)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$
  - Modèle d'observation : $P(Y^n \mid X^n)$

Ce qui donne, modélisé comme un réseau bayésien dynamique :

dBN :

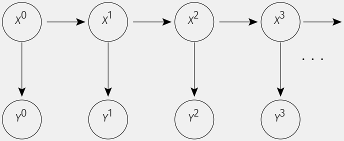


HMM simple

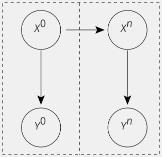
- Une variable d'état discrète (X^n) (à l'instant n).
- Une variable d'observation discrète (Y^n)
- Paramètres du modèle :
  - Condition initiale : $P(X^0)$
  - Modèle de transition : $P(X^n \mid X^{n-1})$
  - Modèle d'observation : $P(Y^n \mid X^n)$

Ce qui donne, modélisé comme un réseau bayésien dynamique :

dBN :



2TBN :



INFÉRENCES DANS LES RÉSEAUX BAYÉSIENS DYNAMIQUES

- A priori, très complexe : nombre de nœuds importants

INFÉRENCES DANS LES RÉSEAUX BAYÉSIENS DYNAMIQUES

- A priori, très complexe : nombre de nœuds importants
- A priori, très complexe : “causes” communes dans un passé (lointain)

- A priori, très complexe : nombre de nœuds importants
- A priori, très complexe : “causes” communes dans un passé (lointain)

Complexité

NP-difficile

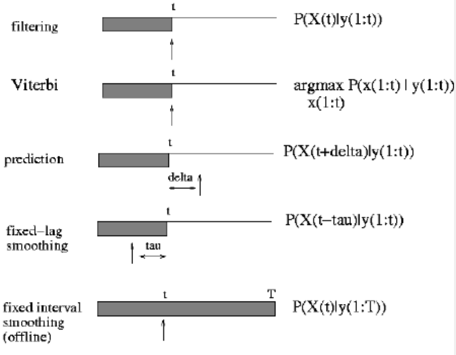
- A priori, très complexe : nombre de nœuds importants
- A priori, très complexe : “causes” communes dans un passé (lointain)

Complexité

NP-difficile

$P(X_t^i \mid y_{[1:r]})$ **?**

- $r = t$: *Filtering*
- $r > t$: *Smoothing*
- $r < t$: *Prediction*
- MPE : *Viterbi*



Soit une base de donnée de séquences $\left(D^{(j,t)}\right)$, où

- j est l'indice de la séquence $j \in J$
- t l'indice du temps dans la séquence $t \in \left\{0, \cdots, T_j\right\}$

Soit une base de donnée de séquences $\left(D^{(j,t)}\right)$, où

- j est l'indice de la séquence $j \in J$
- t l'indice du temps dans la séquence $t \in \left\{0, \cdots, T_j\right\}$

La stationnarité et la propriété de Markov d'ordre 1 nous permettent de spécifier :

Apprentissage d'un dBN

Soit une base de donnée de séquences $\left(D^{(j,t)}\right)$, où

- j est l'indice de la séquence $j \in J$
- t l'indice du temps dans la séquence $t \in \{0, \cdots, T_j\}$

La stationnarité et la propriété de Markov d'ordre 1 nous permettent de spécifier :

Apprentissage d'un dBN

Un dBN s'apprend en deux apprentissages :

Soit une base de donnée de séquences $\left(D^{(j,t)}\right)$, où

- j est l'indice de la séquence $j \in J$
- t l'indice du temps dans la séquence $t \in \left\{0, \cdots, T_j\right\}$

La stationnarité et la propriété de Markov d'ordre 1 nous permettent de spécifier :

Apprentissage d'un dBN

Un dBN s'apprend en deux apprentissages :

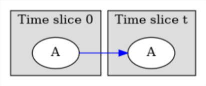
- $P\left(X^{(0)}\right)$ est un BN appris sur la base $\left(D^{(j,0)}\right)_{j\in J}$
- $P\left(X^{(t)} \middle| X^{(t-1)}\right)$ est un BN appris sur la base $\left(D^{(j,t-1)}, D^{(j,t)}\right)_{t\in 1,\cdots,T_j, j\in J}$ sous contrainte d'aucun arc dans $X^{(t-1)}$.

STATIONNARITÉ

Soit une CdM représentant un processus de vieillissement d'un composant A.

STATIONNARITÉ

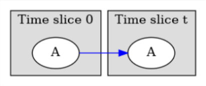
Soit une CdM représentant un processus de vieillissement d'un composant A.



A0	
0	1
0.0000	1.0000

	At	
A0	0	1
0	1.0000	0.0000
1	0.1000	0.9000

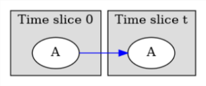
Soit une CdM représentant un processus de vieillissement d'un composant A.



A0	
0	1
0.0000	1.0000

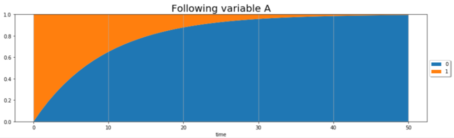
	At	
A0	0	1
0	1.0000	0.0000
1	0.1000	0.9000

Soit une CdM représentant un processus de vieillissement d'un composant *A*.

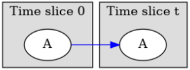


A0	
0	1
0.0000	1.0000

	At	
A0	0	1
0	1.0000	0.0000
1	0.1000	0.9000

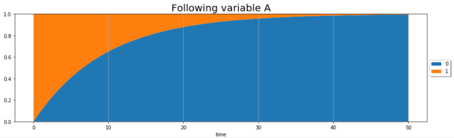


Soit une CdM représentant un processus de vieillissement d'un composant A.



A0	
0	1
0.0000	1.0000

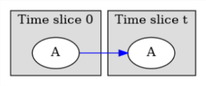
	At	
A0	0	1
0	1.0000	0.0000
1	0.1000	0.9000



Maintenance en $t = 10$: $A^{(10)} = 1$

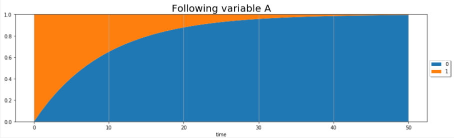
STATIONNARITÉ

Soit une CdM représentant un processus de vieillissement d'un composant A.

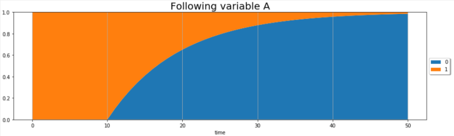


A0	
0	1
0.0000	1.0000

	At	
A0	0	1
0	1.0000	0.0000
1	0.1000	0.9000



Maintenance en $t = 10$: $A^{(10)} = 1$

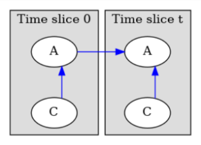


On ne peut pas représenter une intervention sans supprimer la stationnarité du modèle ...

STATIONNARITÉ (2)

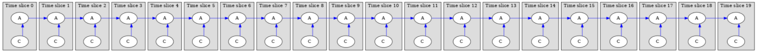
On peut retrouver la stationnarité en introduisant une variable de contrôle C.

On peut retrouver la stationnarité en introduisant une variable de contrôle C.



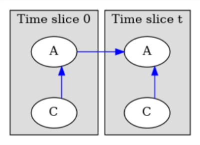
	A0	
C0	0	1
0	0.0000	1.0000
1	0.0000	1.0000

		At	
A0	Ct	0	1
0	0	1.0000	0.0000
1	0	0.1000	0.9000
0	1	0.0000	1.0000
1	1	0.0000	1.0000



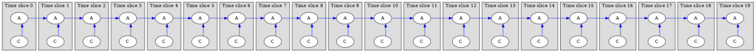
STATIONNARITÉ (2)

On peut retrouver la stationnarité en introduisant une variable de contrôle C.



	A0	
C0	0	1
0	0.0000	1.0000
1	0.0000	1.0000

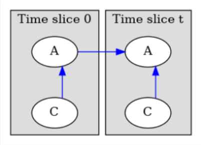
		At	
A0	Ct	0	1
0	0	1.0000	0.0000
1	0	0.1000	0.9000
0	1	0.0000	1.0000
1	1	0.0000	1.0000



Maintenance en $t = 10$: $C^{(10)} = 1$ sinon $C^{(t)} = 0$

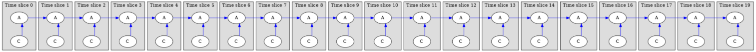
STATIONNARITÉ (2)

On peut retrouver la stationnarité en introduisant une variable de contrôle C.

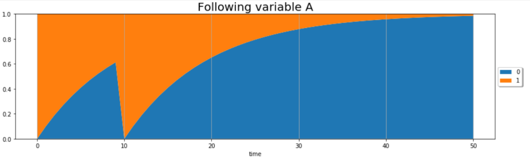


	A0	
C0	**0**	**1**
0	0.0000	1.0000
1	0.0000	1.0000

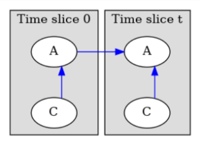
		At	
A0	**Ct**	**0**	**1**
0	**0**	1.0000	0.0000
1	**0**	0.1000	0.9000
0	**1**	0.0000	1.0000
1	**1**	0.0000	1.0000



Maintenance en $t = 10$: $C^{(10)} = 1$ sinon $C^{(t)} = 0$

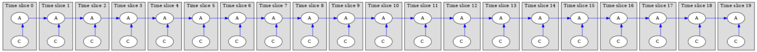


On peut retrouver la stationnarité en introduisant une variable de contrôle C.

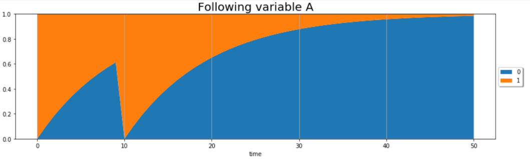


	A0	
C0	**0**	**1**
0	0.0000	1.0000
1	0.0000	1.0000

		At	
A0	**Ct**	**0**	**1**
0	**0**	1.0000	0.0000
1		0.1000	0.9000
0	**1**	0.0000	1.0000
1		0.0000	1.0000

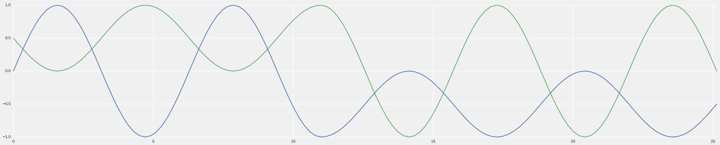


Maintenance en $t = 10$: $C^{(10)} = 1$ sinon $C^{(t)} = 0$

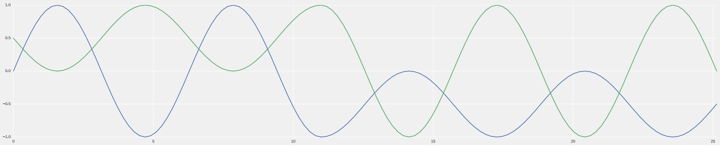


Le coût des variables de contrôle peut être prohibitif. On aimerait un modèle dynamique non stationnaire ...

MOTIVATIONS

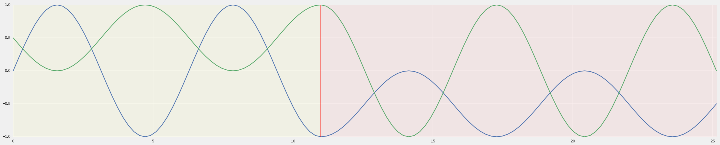


MOTIVATIONS

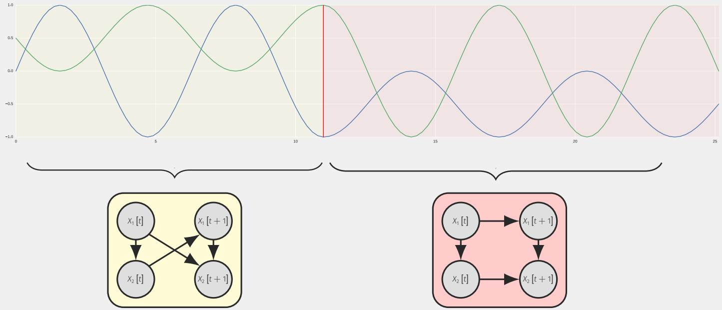


- Streamed data (online)

MOTIVATIONS

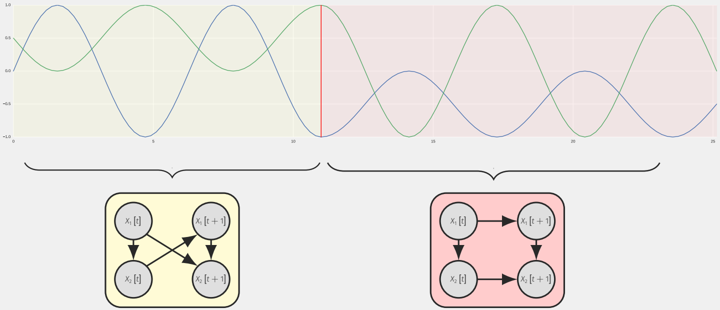


- Streamed data, non i.i.d.
- Non homogène



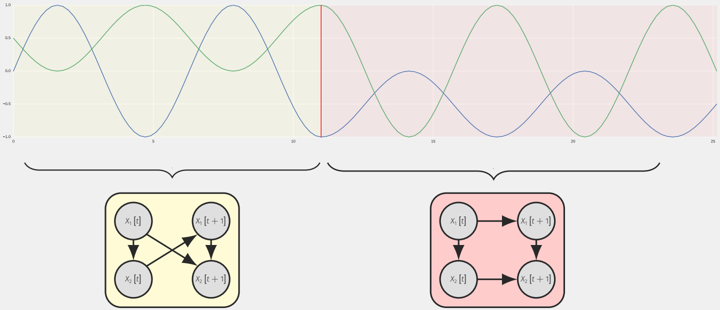
- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental

MOTIVATIONS



- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$

MOTIVATIONS

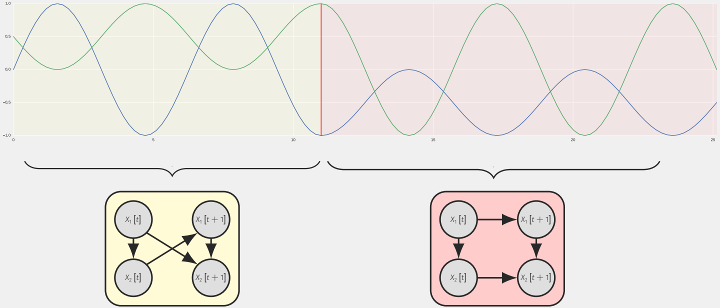


Outline

- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$
- Applications: computer security, bank fraud

- DBNs , ns-DBNs

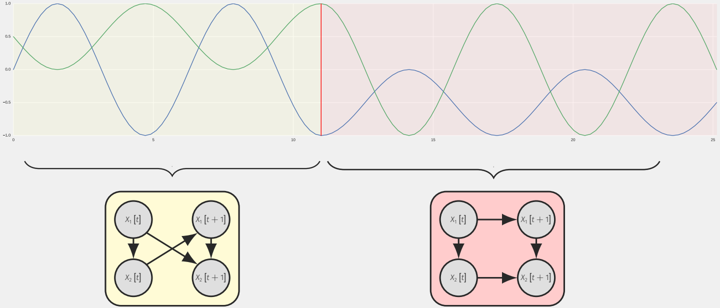
MOTIVATIONS



- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$
- Applications: computer security, bank fraud

Outline

- DBNs , ns-DBNs

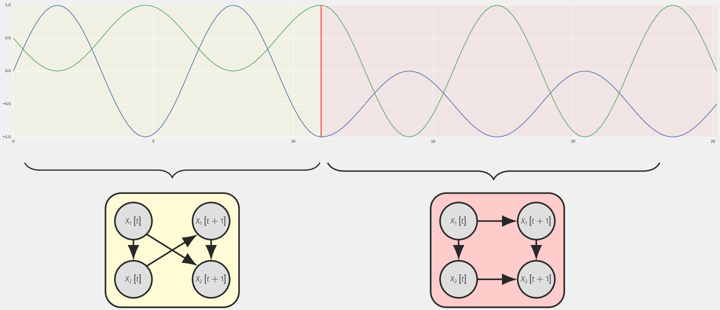


- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$
- Applications: computer security, bank fraud

Outline

- DBNs , ns-DBNs
- Identification et apprentissage

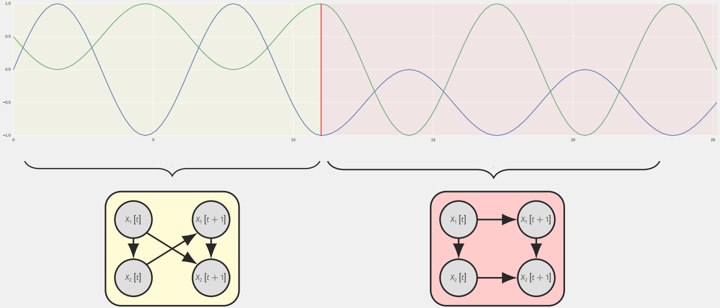
MOTIVATIONS



- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$
- Applications: computer security, bank fraud

Outline

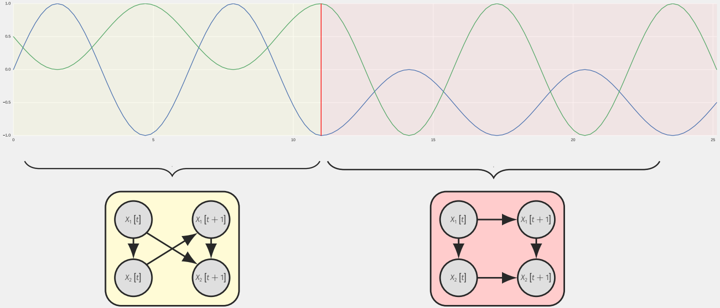
- DBNs , ns-DBNs
- Identification et apprentissage
- Expériences - fenêtres statiques



- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$
- Applications: computer security, bank fraud

Outline

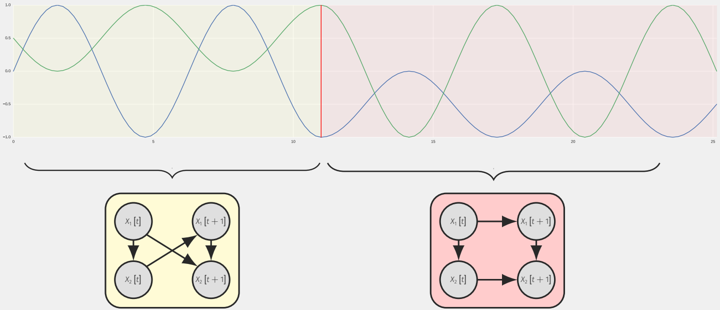
- DBNs , ns-DBNs
- Identification et apprentissage
- Expériences - fenêtres statiques
- Change-point



- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$
- Applications: computer security, bank fraud

Outline

- DBNs , ns-DBNs
- Identification et apprentissage
- Expériences - fenêtres statiques
- Change-point
- Expériences - fenêtres dynamiques

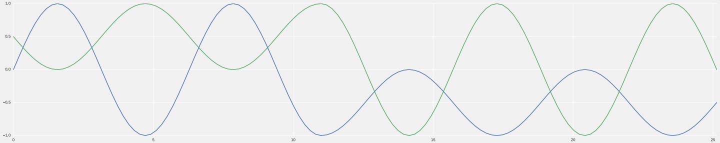


- Streamed data, non i.i.d.
- Non homogène
- Modèle probabiliste dynamique
- Identification DBN et apprentissage incrémental
- $freq\,(change) \ll freq\,(sampling)$
- Applications: computer security, bank fraud

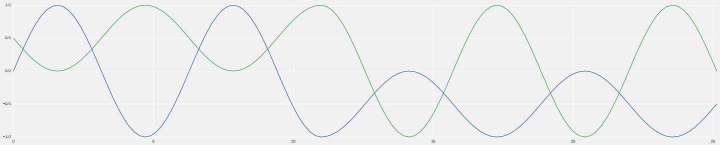
Outline

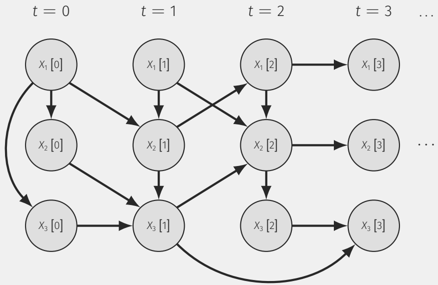
- DBNs , ns-DBNs
- Identification et apprentissage
- Expériences - fenêtres statiques
- Change-point
- Expériences - fenêtres dynamiques
- Conclusion, logs apache

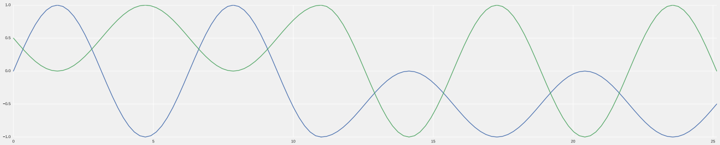
QU'EST-CE QU'UN DBN ?

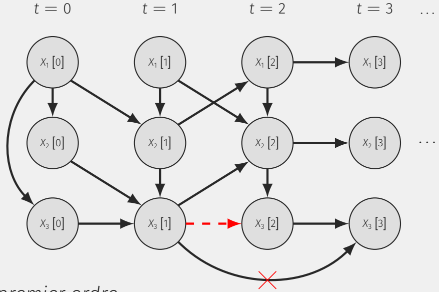


QU'EST-CE QU'UN DBN ?





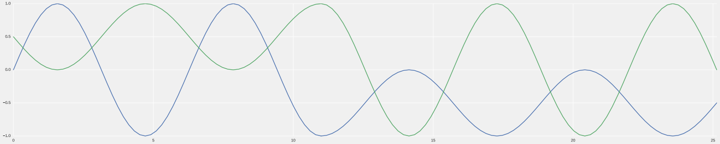


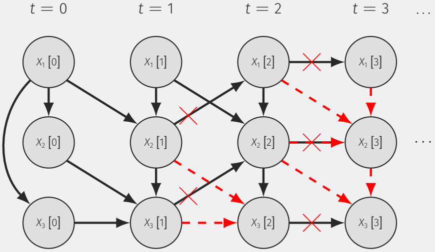


- *propriété de Markov du premier ordre*

$$\mathrm{Pa}\left(X_i\left[t\right]\right) \subset \mathbf{X}\left[t-1\right] \cup \mathbf{X}\left[t\right] \setminus \left\{X_i\left[t\right]\right\}$$

QU'EST-CE QU'UN DBN ?



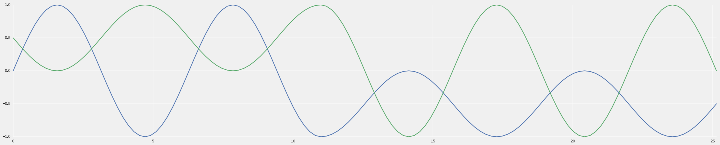


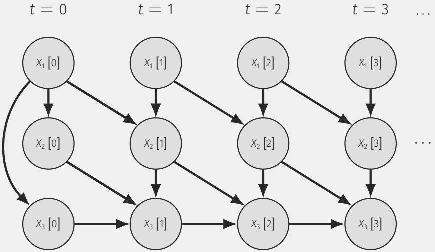
- *propriété de Markov du premier ordre*

$$\mathrm{Pa}\left(X_i\left[t\right]\right) \subset \mathbf{X}\left[t-1\right] \cup \mathbf{X}\left[t\right] \setminus \left\{X_i\left[t\right]\right\}$$

- *homogénéité*

$$\forall t, t' \in [\![1, \tau]\!] : P_i^t = P_i^{t'}$$



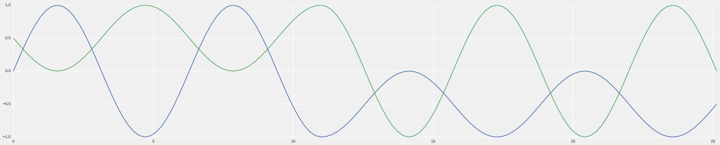


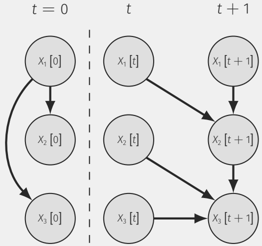
- *propriété de Markov du premier ordre*

$$\mathbf{Pa}\left(X_i\,[t]\right) \subset \mathbf{X}\,[t-1] \cup \mathbf{X}\,[t] \setminus \{X_i\,[t]\}$$

- *homogénéité*

$$\forall t, t' \in [\![1, \tau]\!] : P_i^t = P_i^{t'}$$

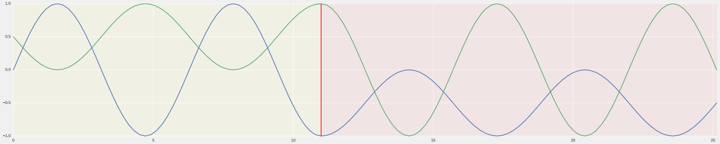


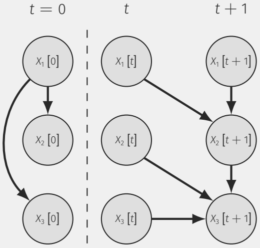


Représentation compacte de P

$$P\left(\mathrm{X}[1]\ldots,\mathrm{X}[\tau]\right) = \prod_{i=1}^{n}\prod_{t=1}^{\tau} P\left(X_i[t] \mid \mathrm{Pa}\left(X_i[t]\right)\right)$$

QU'EST-CE QU'UN DBN ?



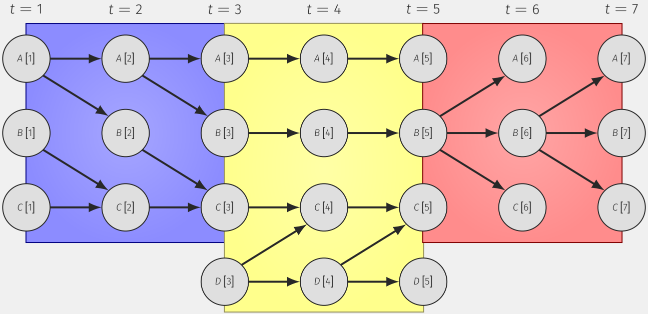


Représentation compacte de P

$$P(\mathrm{X}[1] \ldots, \mathrm{X}[\tau]) = \prod_{i=1}^{n} \prod_{t=1}^{\tau} P(X_i[t] \mid \mathbf{Pa}(X_i[t]))$$

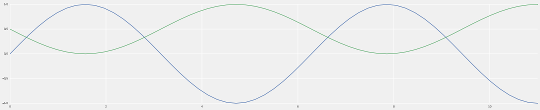
QU'EST-CE QU'UN NS-DBN ?

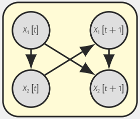
Non-stationary dynamic bayesian network.



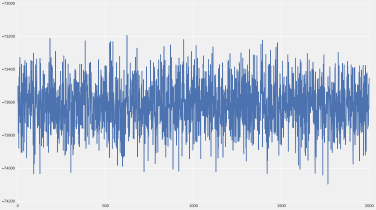
- Perte d'homogénéité sur l'ensemble du processus
- Spécifier $P\left(X_i\left[t\right] \mid \mathbf{Pa}\left(X_i\left[t\right]\right)\right)$ (homogène) pour chaque époque
- Homogénéité par morceaux (algorithmes efficaces)

IDENTIFIER ÉPOQUES ET DBN ASSOCIÉ À CHACUNE

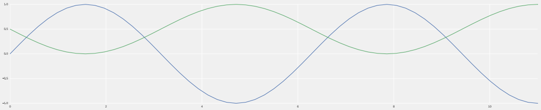


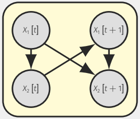


$$\mathcal{LL}_i\left(\mathrm{w}, DBN_i\right)$$

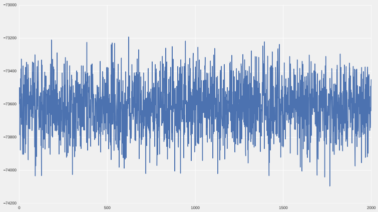


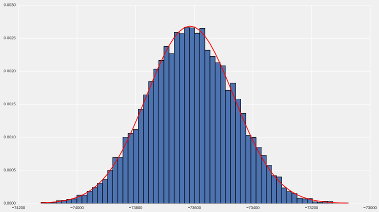
IDENTIFIER ÉPOQUES ET DBN ASSOCIÉ À CHACUNE

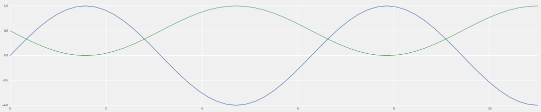


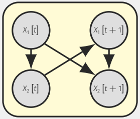


$$\mathcal{LL}_i\left(\mathrm{w}, DBN_i\right) \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

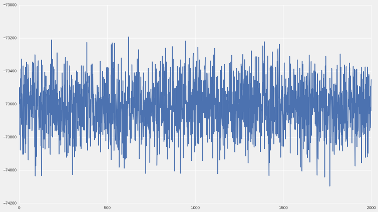


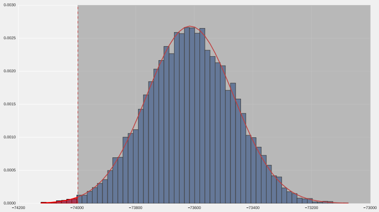


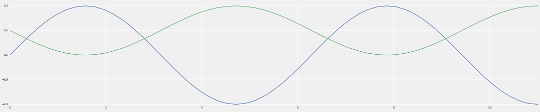


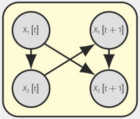


$$\mathcal{LL}_i\left(\mathrm{w}, DBN_i\right) \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

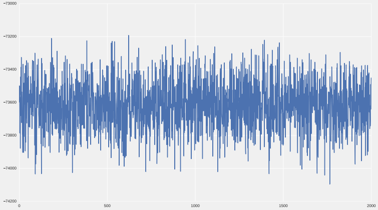


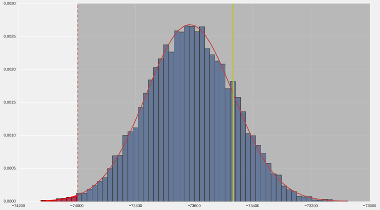




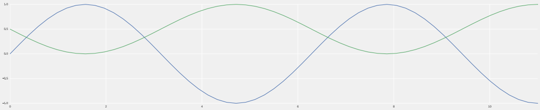


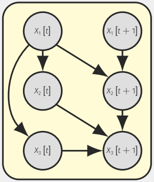
$$\mathcal{LL}_i\left(\mathrm{w}, DBN_i\right) \sim \mathcal{N}\left(\mu_i, \sigma_i^2\right)$$

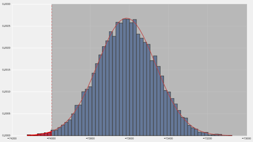


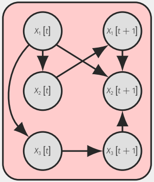


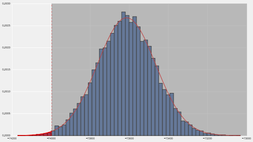
IDENTIFIER ÉPOQUES ET DBN ASSOCIÉ À CHACUNE

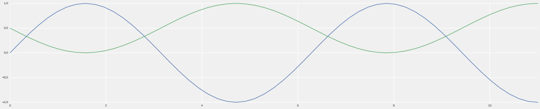


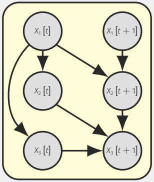


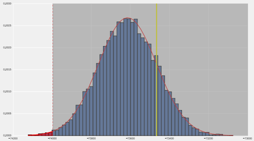




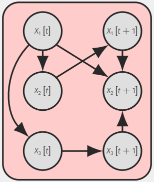


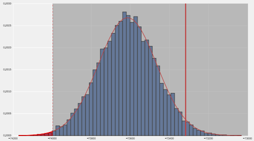




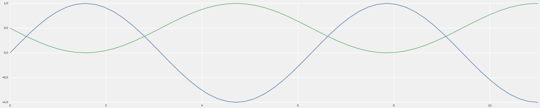


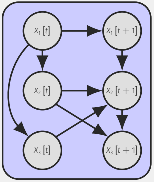
$$\underset{i}{\operatorname{argmax}} \mathcal{LL}_i\left(w, DBN_i\right)$$

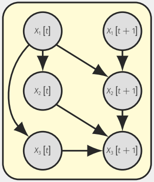


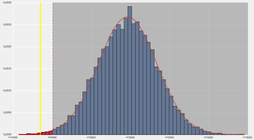


IDENTIFIER ÉPOQUES ET DBN ASSOCIÉ À CHACUNE

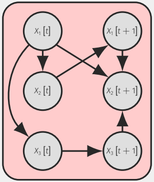


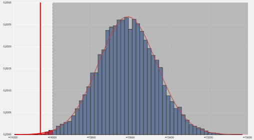






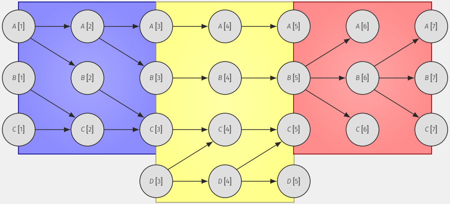
$$\operatorname*{argmax}_i \mathcal{LL}_i\left(w, DBN_i\right)$$

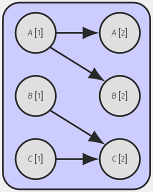




QUANTES

Learning with new and/or missing variables.

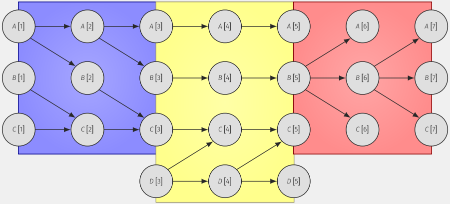


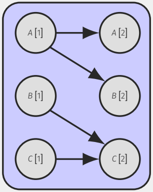


$A[t]$	$B[t]$	$C[t]$	$D[t]$
0	1	5	4
3	6	9	5
8	2	12	3
1	5	3	8
6	7	5	1
3	0	4	7

QUANTES

Learning with new and/or missing variables.

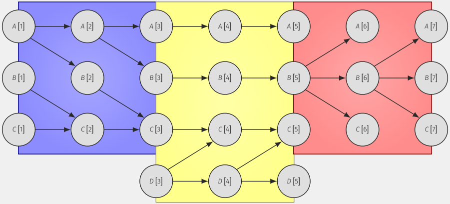


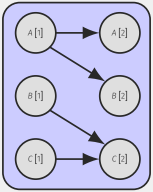


$A[t]$	$B[t]$	$C[t]$	$D[t]$
0	1	5	4
3	6	9	5
8	2	12	3
1	5	3	8
6	7	5	1
3	0	4	7

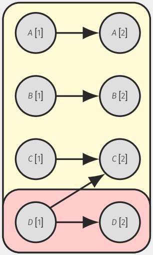
rejeter $D[t]$ dans DB

Learning with new and/or missing variables.





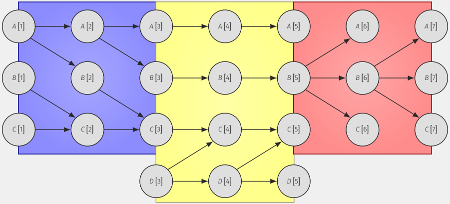
$A[t]$	$B[t]$	$C[t]$	$D[t]$
0	1	5	4
3	6	9	5
8	2	12	3
1	5	3	8
6	7	5	1
3	0	4	7

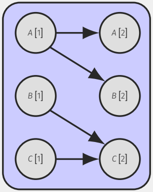


$A[t]$	$B[t]$	$C[t]$
0	1	5
3	6	9
8	2	12
1	5	3
6	7	5
3	0	4

rejeter $D[t]$ dans DB

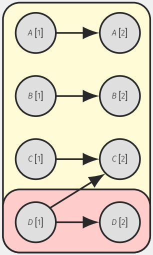
Learning with new and/or missing variables.





$A[t]$	$B[t]$	$C[t]$	$D[t]$
0	1	5	4
3	6	9	5
8	2	12	3
1	5	3	8
6	7	5	1
3	0	4	7

rejeter $D[t]$ dans DB

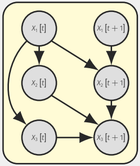


$A[t]$	$B[t]$	$C[t]$
0	1	5
3	6	9
8	2	12
1	5	3
6	7	5
3	0	4

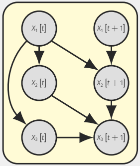
calculer $P(C[t+1] \mid C[t])$

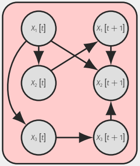
$$\sum_D P(C[t+1] \mid C[t], D[t]) \cdot P(D[t])$$

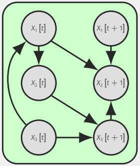
EXPÉRIENCES - GÉNÉRATION DB

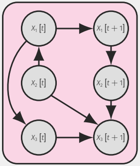


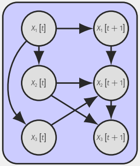
EXPÉRIENCES - GÉNÉRATION DB



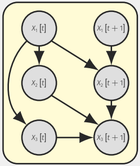




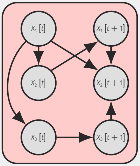




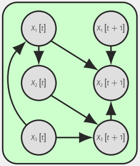
EXPÉRIENCES - GÉNÉRATION DB



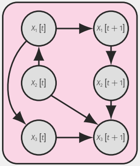
$X_1[t]$	$X_2[t]$	$X_3[t]$
0	1	5
3	6	9
	...	



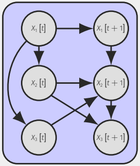
$X_1[t]$	$X_2[t]$	$X_3[t]$
8	2	12
1	5	3
	...	



$X_1[t]$	$X_2[t]$	$X_3[t]$
3	7	4
6	1	9
	...	

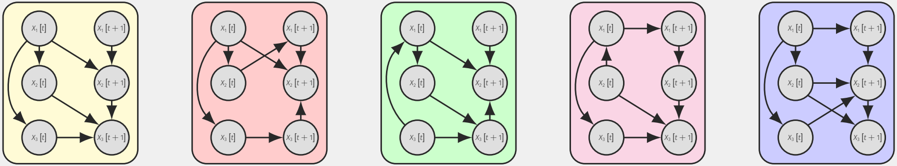


$X_1[t]$	$X_2[t]$	$X_3[t]$
3	9	2
8	6	1
	...	



$X_1[t]$	$X_2[t]$	$X_3[t]$
11	3	9
7	5	6
	...	

EXPÉRIENCES - GÉNÉRATION DB



$X_1[t]$	$X_2[t]$	$X_3[t]$
0	1	5
3	6	9
	...	

$X_1[t]$	$X_2[t]$	$X_3[t]$
8	2	12
1	5	3
	...	

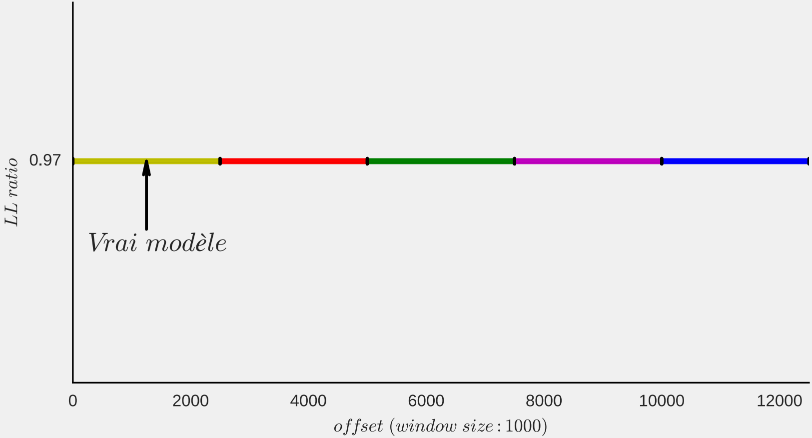
$X_1[t]$	$X_2[t]$	$X_3[t]$
3	7	4
6	1	9
	...	

$X_1[t]$	$X_2[t]$	$X_3[t]$
3	9	2
8	6	1
	...	

$X_1[t]$	$X_2[t]$	$X_3[t]$
11	3	9
7	5	6
	...	

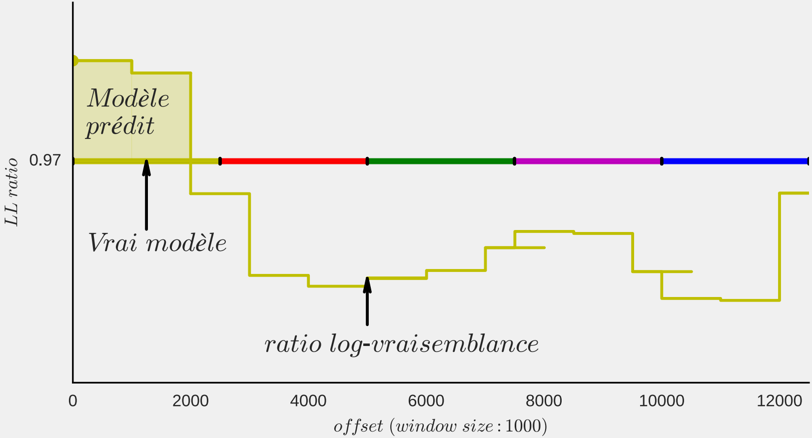
$X_1[t]$	$X_2[t]$	$X_3[t]$
0	1	5
	...	
8	2	12
	...	
3	7	4
	...	

EXPÉRIENCES - ÉVALUATION

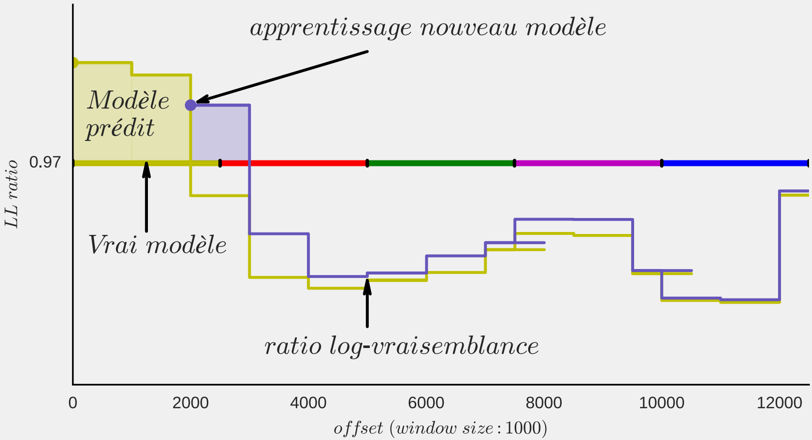


Comment interpréter les expériences.

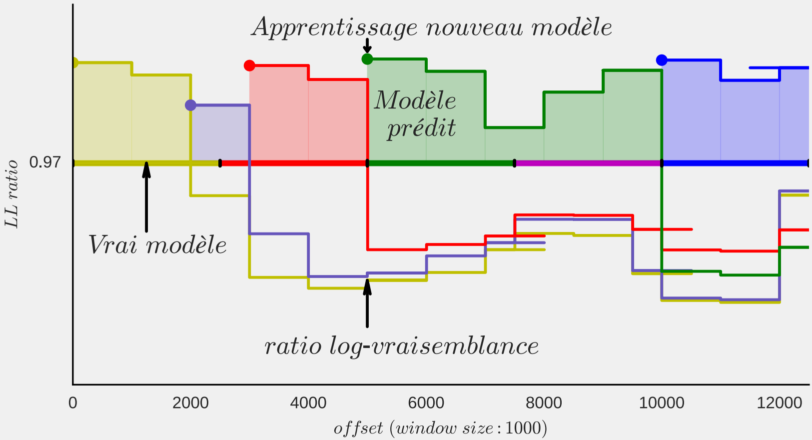
EXPÉRIENCES – ÉVALUATION



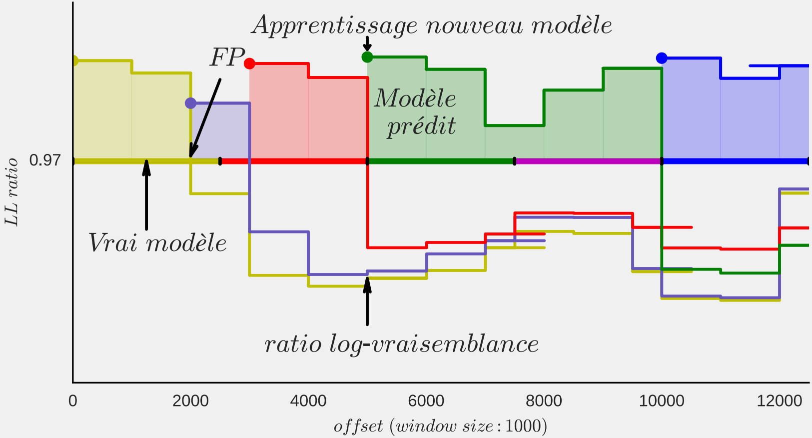
Comment interpréter les expériences.



Comment interpréter les expériences.

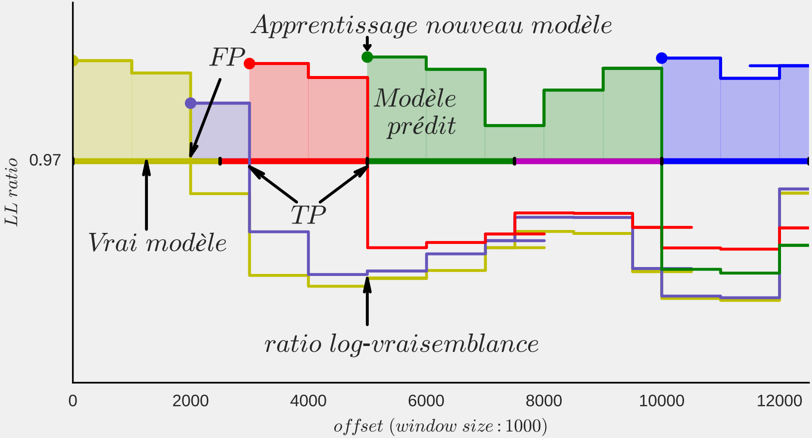


Comment interpréter les expériences.



Comment interpréter les expériences.

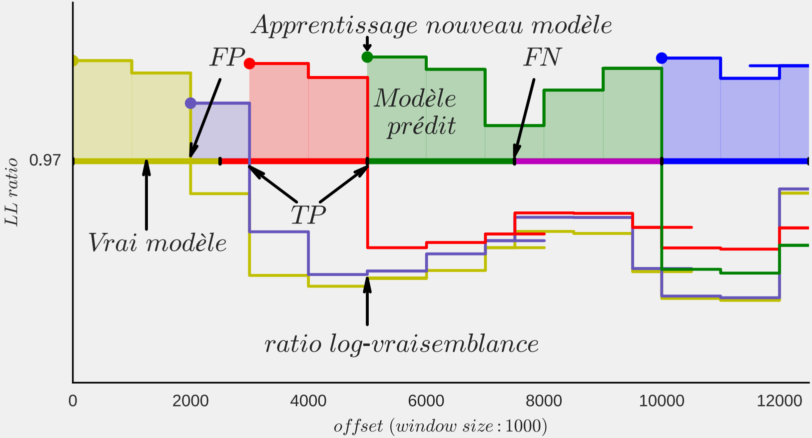
FP : Faux positif (transition erronée)



Comment interpréter les expériences.

FP : Faux positif (transition erronée)

TP : Vrai positif (transition correcte)

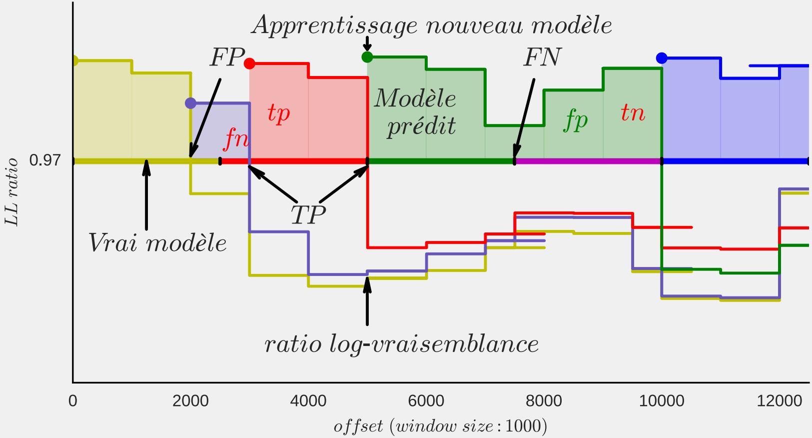


Comment interpréter les expériences.

FP : Faux positif (transition erronée)

FN : Faux négatif (transition ratée)

TP : Vrai positif (transition correcte)



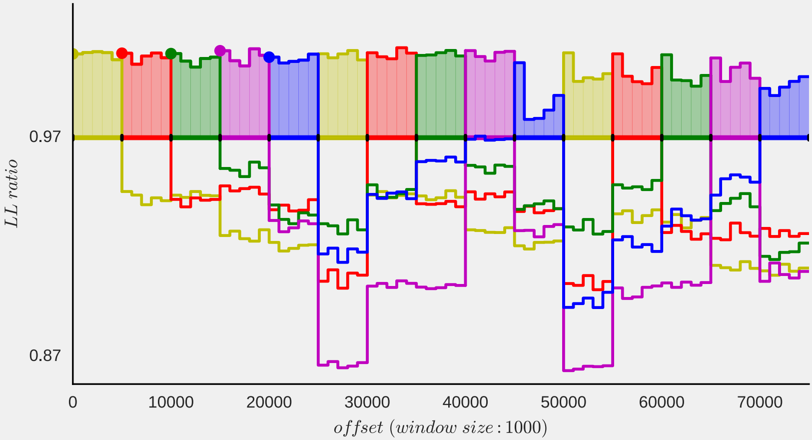
Comment interpréter les expériences.

FP : Faux positif (transition erronée)

TP : Vrai positif (transition correcte)

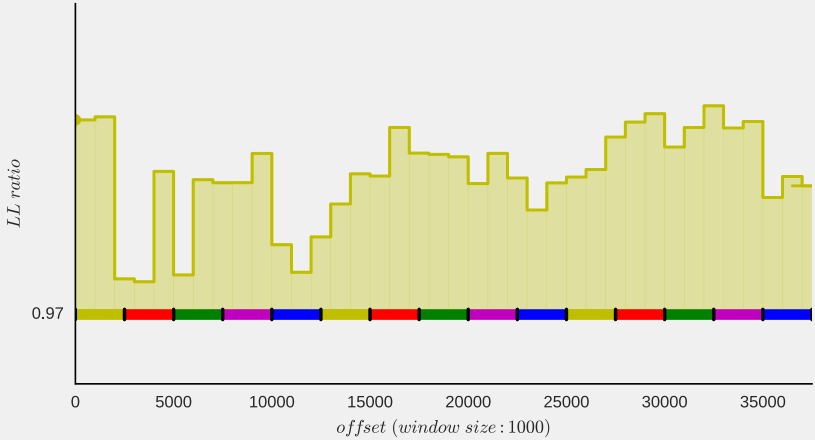
FN : Faux négatif (transition ratée)

fp, *tp*, *fn*, *tn* : as above, for events



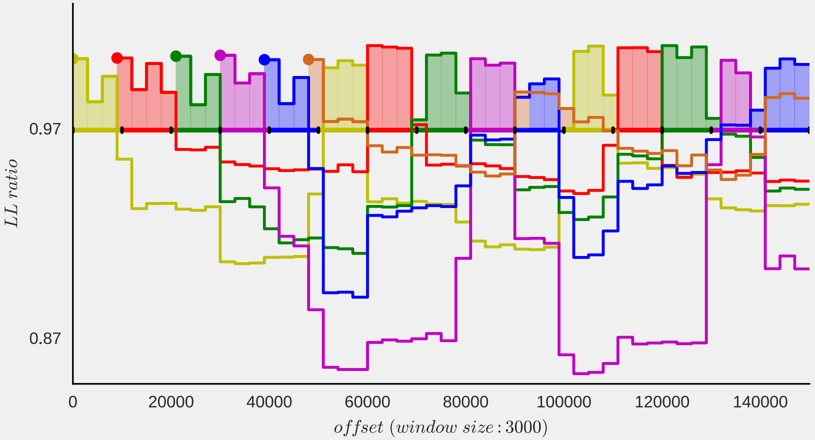
Résultats pour des époques de 5K observations, *Hellinger* < 0.8, fenêtres statiques

EXPÉRIENCES - FENÊTRE STATIQUE

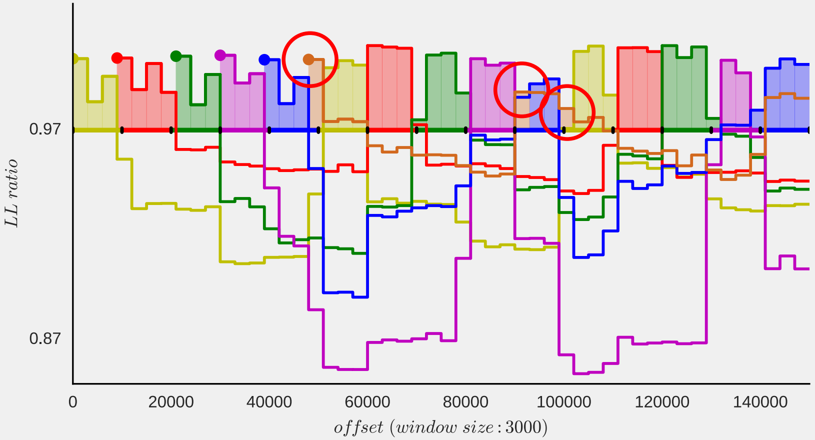


Résultats pour des époques de 2*K*5 observations, *Hellinger* < 0.8, fenêtres statiques

EXPÉRIENCES - FENÊTRE STATIQUE



Résultats pour des époques de 10K observations, *Hellinger* < 0.8, fenêtres statiques

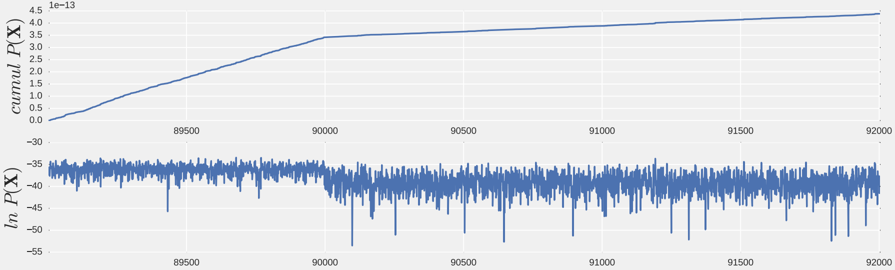


Résultats pour des époques de 10K observations, *Hellinger* < 0.8, fenêtres statiques

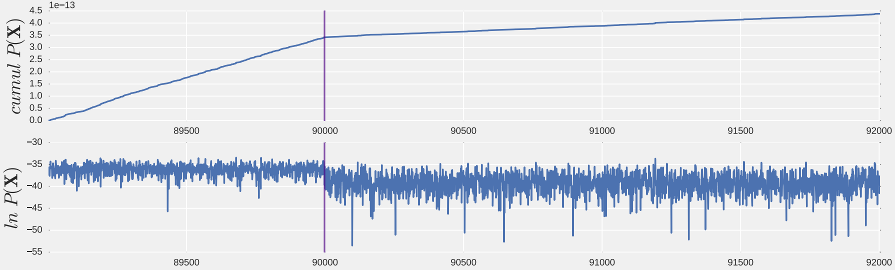
RÉSULTATS SYNTHÉTIQUES - FENÊTRE STATIQUE

epoch	*window size*	*FN*	*FP*	*TP*	*average error*
2500	1000	0.5	0.0	0.5	251.046
2500	2000	0.602	0.0	1.0	521.052
5000	1500	0.101	0.035	0.965	351.216
5000	3000	0.0	0.06	0.94	752.1
10000	1500	0.0	0.131	0.869	381.356
10000	3000	0.1017	0.0083	0.992	772.81
15000	2000	0.0	0.204	0.795	512.82

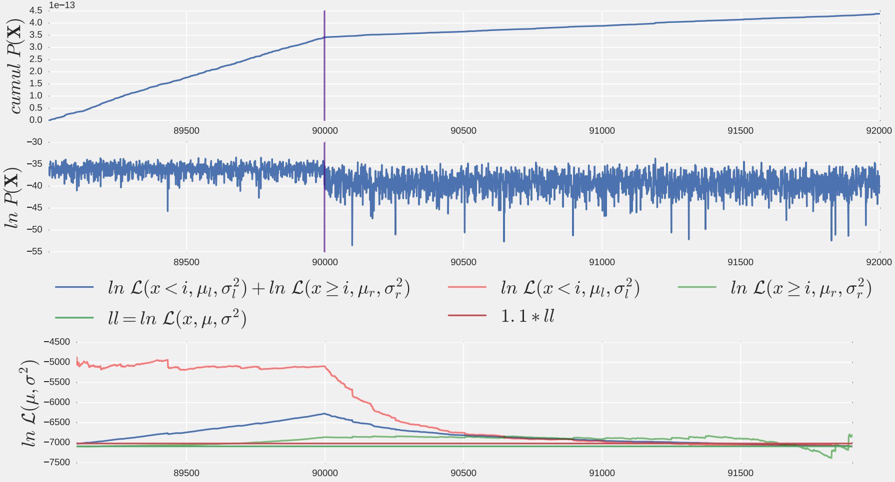
CHANGE-POINT



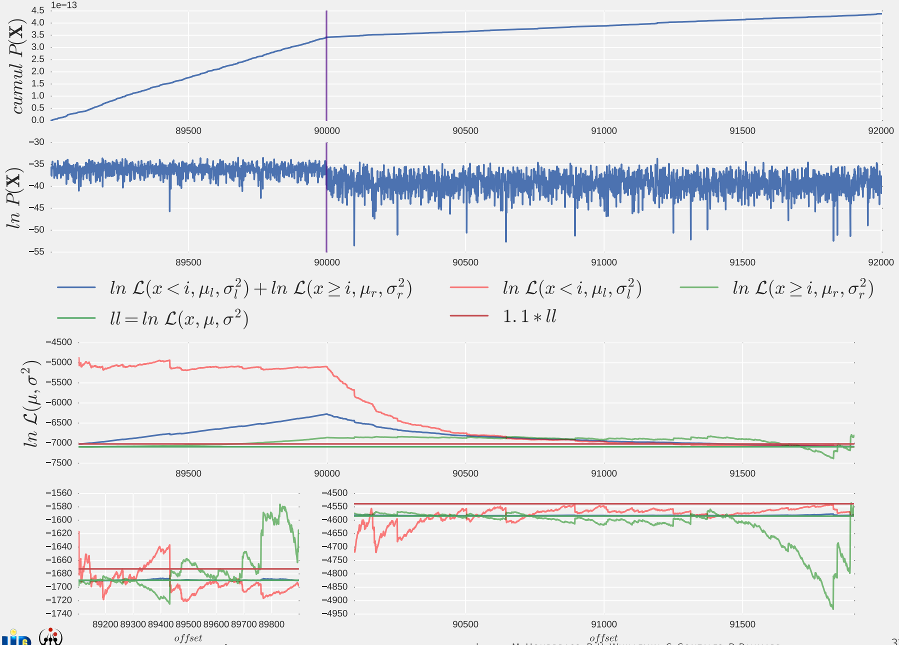
CHANGE-POINT



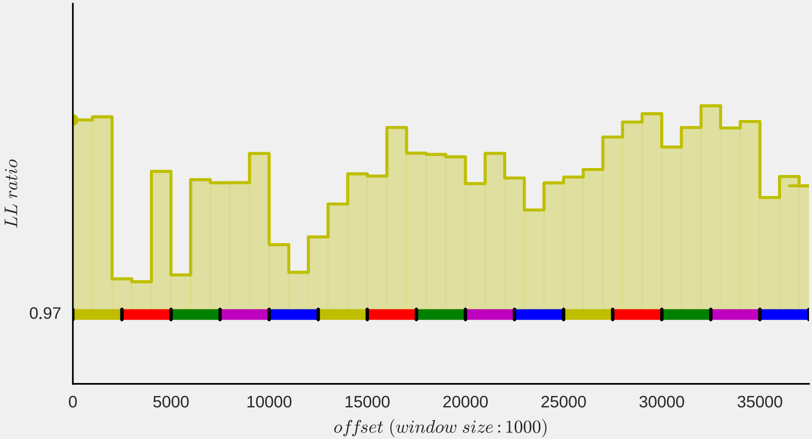
CHANGE-POINT



CHANGE-POINT

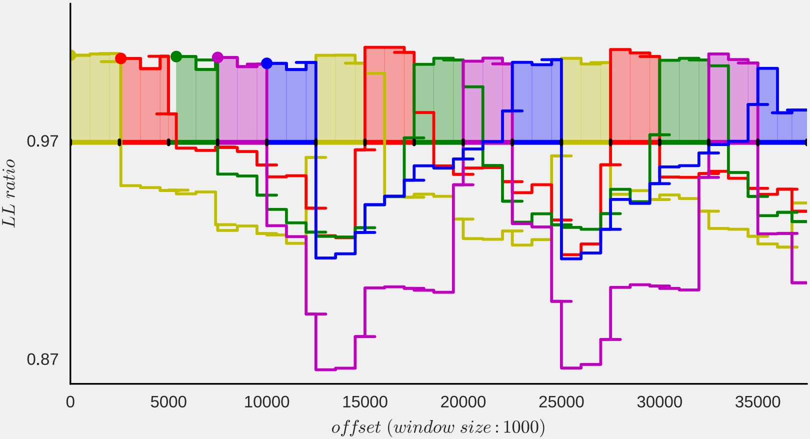


EXPÉRIENCES - FENÊTRE DYNAMIQUE



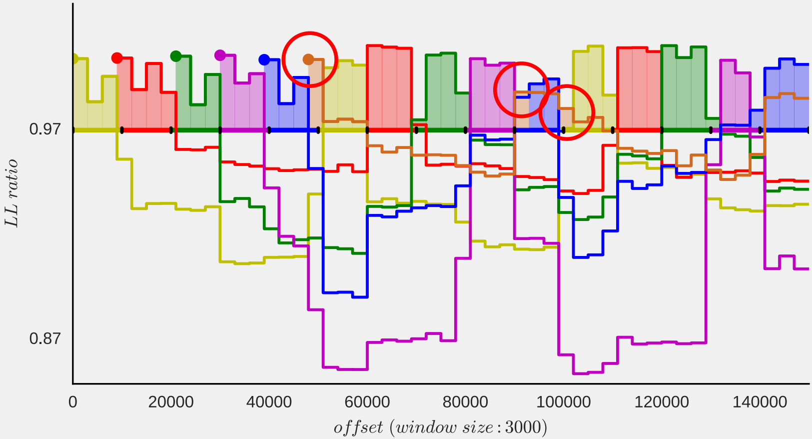
Résultats pour des époques de 2K5 observations, *Hellinger* < 0.8, fenêtres statiques

EXPÉRIENCES - FENÊTRE DYNAMIQUE



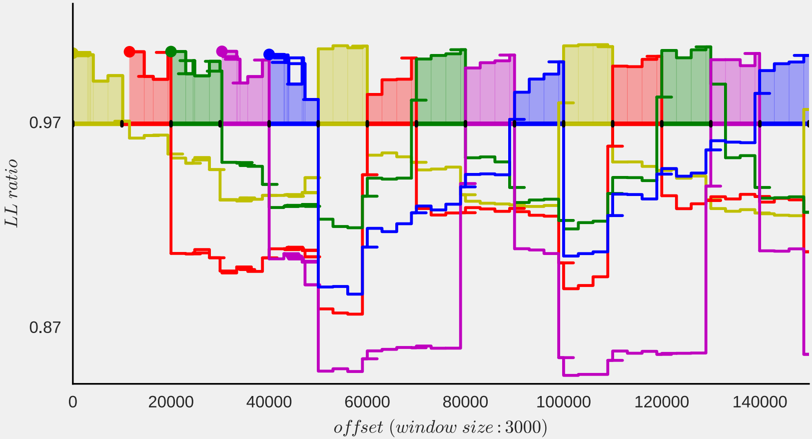
Résultats pour des époques de 2*K*5 observations, *Hellinger* < 0.8, fenêtres dynamiques

EXPÉRIENCES - FENÊTRE DYNAMIQUE



Résultats pour des époques de 10K observations, *Hellinger* < 0.8, fenêtres statiques

EXPÉRIENCES - FENÊTRE DYNAMIQUE



Résultats pour des époques de 10*K* observations, *Hellinger* < 0.8, fenêtres dynamiques

RÉSULTATS SYNTHÉTIQUES - FENÊTRE DYNAMIQUE

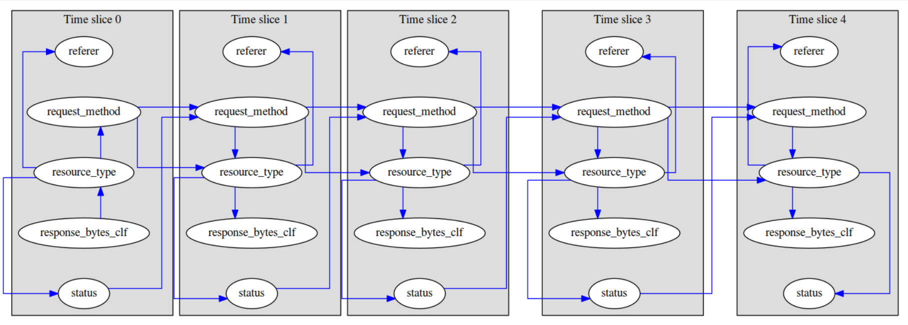
epoch	window size	FN	FP	TP	average error
2500	1000	0.5	0.0	0.5	251.046
2500	2000	0.602	0.0	1.0	521.052
5000	1500	0.101	0.035	0.965	351.216
5000	3000	0.0	0.06	0.94	752.1
10000	1500	0.0	0.131	0.869	381.356
10000	3000	0.1017	0.0083	0.992	772.81
15000	2000	0.0	0.204	0.795	512.82

FN	FP	TP	average error
0.0	0.0	1.0	4.399
0.0	0.0	1.0	2.435
0.0	0.0	1.0	3.0966
0.0	0.0	1.0	8.702
0.0	0.0	1.0	32.923
0.0	0.0	1.0	19.559
0.0	0.0	1.0	8.551

CONCLUSIONS ET TRAVAUX FUTURS

- Réseaux bayésiens dynamiques non-homogènes
  - Comme une collection de modèles avec structures et/ou paramètres différents
  - Homogénéité par morceaux
- Identification correcte du nombre de transitions, de leur positions et des modèles
- Évènements singuliers et outliers ?
- Quel serait le modèle de transition entre époques ?

173.36.240.171 — —[01/Jun/2015 : 00 : 01 : 19 + 0000]" GET/projects/scapy/HTTP/1.1" 2006205" http : //r.search.yahoo.com/$_y$lt = AwrTccnLoGtVzh8A$_N$UnnIlQ;$_y$ lu = X3oDMTByYnR1Zmd1BGNvbG8DZ3ExBHBvcwMyBHZ0aWQDBHNlYWNzcg — —/RV = 2/RE = 1433145676/RO = 10/RU = http%3a%2f%2fsecdev.org%2fprojects%2fscapy%2f/RK = 0/RS = DV3KSkU3cpgfOsdls5DOVKn1zrs — "" Mozilla/5.0(WindowsNT6.1; WOW64; rv : 38.0)Gecko/20100101Firefox/38.0"



LOGS APACHE

- 1 mois de logs - 29341 adresse ip, quelques anomalies / patterns d'attaque(5)
- 33 modèles découverts
- Pour chaque ip, un set de modèles
- Un modèle apparaît seulement pendant les attaques et inclut tous les patterns connus