SPLEX
Statistiques pour la classification et fouille de données en génomique

Réseaux bayésiens
Apprentissage

Pierre-Henri WUILLEMIN

DEcision, Système Intelligent et Recherche opérationnelle
LIP6
pierre-henri.wuillemin@lip6.fr
http://webia.lip6.fr/~phw/splex

Une base de données

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

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

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

Vers un réseau bayésien (1) : χ^2

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

Soit X et Y deux v.a. binaires,
si $X \perp\!\!\!\perp Y$ alors $\forall i,j,\ p(X=i,Y=j) = p(X=i)\cdot p(Y=j)$

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

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

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

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

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

Vers un réseau bayésien (2) : tableau de contingence

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

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

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

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

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

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

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

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

Vers un BN (5) : liste d'indépendances

- $A \perp\!\!\!\perp C \mid B$
- $A \perp\!\!\!\perp D \mid B$
- $C \perp\!\!\!\perp D \mid B$

Un réseau bayésien

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

Calcul dans un réseau bayésien

Les quelques manipulations de base :

Chain rule $$P(X_1, \cdots, X_n) = \prod_{i=1}^{n} P\left(X_i \mid X_1, \cdots, X_{i-1}\right)$$

Markov local $$P(X_1, \cdots, X_n) = \prod_{i=1}^{n} P\left(X_i \mid \text{parents}(X_i)\right)$$

Marginalisation $$\sum_{y} P(x, y \mid z) = p(x \mid z)$$

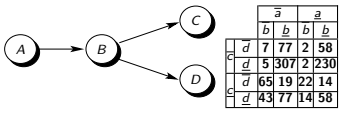
Somme totale $$\sum_{y} P(y \mid z) = 1$$

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

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

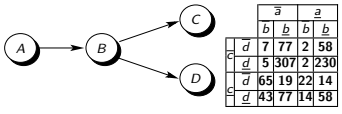
Loi de Bayes $$P(x \mid y, z) \propto P(y \mid x, z) \cdot P(x \mid z)$$

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



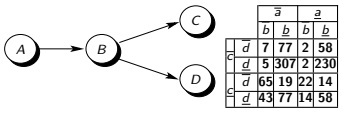
		$\overline{a}$		$\underline{a}$	
		$\overline{b}$	$\underline{b}$	$\overline{b}$	$\underline{b}$
$\overline{c}$	$\overline{d}$	7	77	2	58
	$\underline{d}$	5	307	2	230
$\underline{c}$	$\overline{d}$	65	19	22	14
	$\underline{d}$	43	77	14	58

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



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

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



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

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 ($P(0 \mid Y = 1)$?)

La base de données peut être :

- complète,
- incomplète.

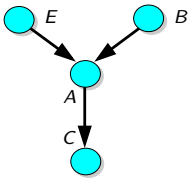
Les connaissances a priori sont très variables ; par exemple :

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

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

Apprentissage des paramètres, données complètes

$$D : \begin{bmatrix} d_1^A & d_1^B & d_1^C & d_1^E \\ \dots & \dots & \dots & \dots \\ V & F & F & V \\ \dots & \dots & \dots & \dots \\ d_M^A & d_M^B & d_M^C & d_M^E \end{bmatrix}$$



En appelant Θ l'ensemble des paramètres du modèle et $L(\Theta : D)$ la vraisemblance :

$$\begin{aligned} L(\Theta : D) &= P(D \mid \Theta) \\ &= \prod_{m=1}^{M} P(d_m \mid \Theta) \qquad \text{(échantillons indépendants, identiquement distribués)} \\ &= \prod_{m=1}^{M} P(E = d_m^E, B = d_m^B, A = d_m^A, C = d_m^C \mid \Theta) \end{aligned}$$

Apprentissage des paramètres, données complètes (2)

En renommant E, B, A, C par X_1, X_2, X_3, X_4,

$$\begin{aligned} L(\Theta : D) &= \prod_{m=1}^{M} P(X_1 = d_m^1, X_2 = d_m^2, X_3 = d_m^3, X_4 = d_m^4 \mid \Theta) \\ &= \prod_{m=1}^{M} \prod_{i=1}^{4} P(X_i \mid Pa_i, \Theta) = \prod_{m=1}^{M} \prod_{i=1}^{4} P(X_i \mid Pa_i, \Theta_i) \\ &= \prod_{i=1}^{4} \prod_{m=1}^{M} P(X_i \mid Pa_i, \Theta_i) = \prod_{i=1}^{4} L_i(\Theta_i : D) \end{aligned}$$

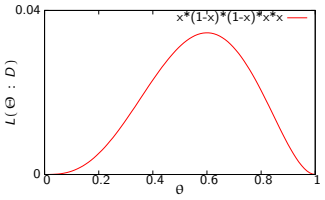
L'estimation des paramètres d'un réseau bayésien se décomposent en l'estimation des paramètres de chaque loi de probabilité conditionnelle

Maximisation de la vraisemblance

Soit une variable binaire X. Avec $\theta = P(X = 1)$:

$$\Theta = \{\theta, 1-\theta\}$$
$$D = (1, 0, 0, 1, 1)$$
$$L(\Theta : D) = \prod_m P(X = d_m \mid \Theta)$$

Ici : $L(\Theta : D) = \theta \cdot (1-\theta) \cdot (1-\theta) \cdot \theta \cdot \theta$.



$$\theta^* = \text{argmax}_\theta \left(\theta^3 \cdot (1-\theta)^2\right)$$

Variable multinomiale

Pour X v.a. de valeurs $(x_1, \cdots, x_n)$,
avec $\Theta = (\theta_1, \cdots, \theta_n)$ où $\theta_i = P(X = x_i)$,
et N_i est le nombre d'occurence de x_i dans D, on a :

$$L(\Theta : D) = \prod_{i=1}^{n} \theta_i^{N_i} \qquad \text{et} \qquad \Theta^* = \text{argmax}_\Theta \left(L(\Theta : D)\right)$$

Prédiction bayésienne

$\theta_i = P(X = i)$ suit une distribution $P(\theta_i \mid D)$.

On peut alors estimer $P(X = i \mid D)$ comme l'espérance mathématique de θ_i :

Prédiction bayésienne

$$P(X = i \mid D) = \int_{\theta_i} \theta_i \cdot P(\theta_i \mid D)\, d\theta_i$$
$$\text{avec } P(\theta \mid D) \propto L(\theta : D) \cdot P(\theta)$$

Cette méthode permet de prendre en compte un *a priori* sur Θ ; par exemple, pour intégrer des connaissances d'expert ou pour rendre plus stable les estimations avec un petit échantillon D.

Apprentissage des paramètres, données complètes (3)

À partir de la base, on peut calculer $N(x_1, x_2, x_3, x_4)$. On note alors $N_{i,j,k}$ le nombre de fois où la variable X_i a pris la valeur k et ses parents la valeur (t-uple) j.

Estimation des paramètres

Deux méthodes possibles pour l'estimation des paramètres :

- MLE (Maximum Likelihood Estimation)

$$\hat{\theta}_{\{x_i|pa_i\}} = \frac{N_{i,j,k}}{N_{i,j}}$$

- Estimation bayésienne (avec *a priori de Dirichlet)*

$$\hat{\theta}_{\{x_i|pa_i\}} = \frac{\alpha_{i,j,k} + N_{i,j,k}}{\alpha_{i,j} + N_{i,j}}$$

Les $\alpha(\cdot)$ correspondents donc aux *a priori* que l'on intègre dans le modèle. La stabilisation s'observe, par exemple, en supposant (à l'extrême) $N(pa_i) = 0$: pas de cas pour cette estimation dans la base.

Les 2 estimations sont consistantes et équivalentes asymptotiquement.

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 :

Espace des réseaux bayésiens

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

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

La recherche exhaustive n'est pas possible. L'espace est bien trop grand.

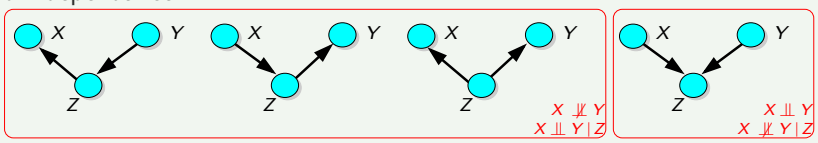
Apprentissage de structure - introduction

Tableau général de l'apprentissage

- Recherche de relation symétrique + orientation (*causalité*)
  - algorithme IC/PC
  - algorithme IC*/FCI
- 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.
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)

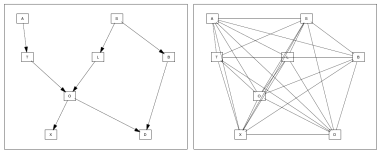
1. Construire le graphe (non orienté) des relations de dépendance trouvées statistiquement (χ^2 ou autre) :
   - Ajouter des arêtes à partir du graphe vide.
   - Retirer des arêtes à partir du graphe complet.
2. Détecter les V-structures et les orientations qu'elles impliquent.
3. Finaliser les orientations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

Écueils principaux :un très grand nombre de tests d'indépendances, chaque test étant très sensible au nombre de données disponibles.

Exemple PC

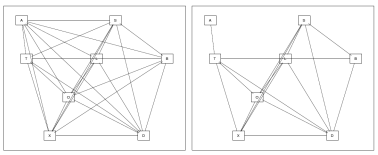
- Soit un réseau bayésien (à gauche) qui a permis de créer une base de 5000 cas. [1]

Etape 0 : Graphe non orienté reliant tous les nœuds.



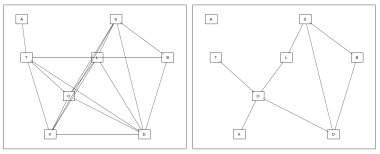
- Par des χ^2, on teste toutes les indépendances marginales ($X \perp\!\!\!\perp Y$) puis les indépendances par rapport à une variable ($X \perp\!\!\!\perp Y \mid Z$).

Etape 1a : Suppression des ind. conditionnelles d'ordre 0



On trouve : $A \perp\!\!\!\perp S$, $L \perp\!\!\!\perp A$, $B \perp\!\!\!\perp A$, $O \perp\!\!\!\perp A$, $X \perp\!\!\!\perp A$, $D \perp\!\!\!\perp A$, $T \perp\!\!\!\perp S$, $L \perp\!\!\!\perp T$,$O \perp\!\!\!\perp B$, $X \perp\!\!\!\perp B$.

Etape 1b : Suppression des ind. conditionnelles d'ordre 1



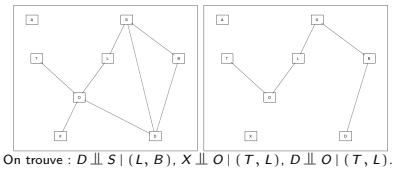
On trouve : $T \perp\!\!\!\perp A \mid O$, $O \perp\!\!\!\perp S \mid L$, $X \perp\!\!\!\perp S \mid L$, $B \perp\!\!\!\perp T \mid S$, $X \perp\!\!\!\perp T \mid O$, $D \perp\!\!\!\perp T \mid O$, $B \perp\!\!\!\perp L \mid S$, $X \perp\!\!\!\perp L \mid O$, $D \perp\!\!\!\perp L \mid O$, $D \perp\!\!\!\perp X \mid O$.

1. Exemple de Philippe Leray

Exemple PC

- On continue les χ^2 d'ordre supérieur

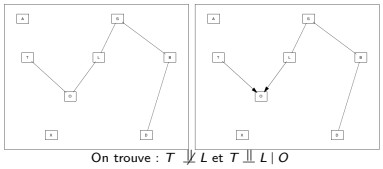
Etape 1c : Suppression des ind. conditionnelles d'ordre 2



On trouve : $D \perp\!\!\!\perp S \mid (L, B)$, $X \perp\!\!\!\perp O \mid (T, L)$, $D \perp\!\!\!\perp O \mid (T, L)$.

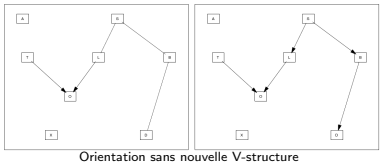
- Recherche des V-Structure, propagation des contraintes d'orientations puis orientations des dernières arêtes en restant Markov-équivalent.

Etape 2 : Recherche des V-structures



On trouve : $T \not\perp\!\!\!\perp L$ et $T \perp\!\!\!\perp L \mid O$

Etape 4 : Instanciation du PDAG



Orientation sans nouvelle V-structure

- Conclusion : avec 5000 cas, PC perd des informations sur des χ^2 faussés.

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** : 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.

Précision sur la décomposition locale

Décomposition

On dira qu'un score $Q(T, \Theta, D)$ est décomposable si $\exists \{q_i, \forall i$ nœud de $T\}$ famille de fonctions telle que

$$Q(T, \Theta, D) = \sum_i q_i(i, pa(i), D[i, pa(i)])$$

Les fonctions q_i dépendent du nœud i, des parents de i et de la partie de la base de donnée qui correspond à ces nœuds.

Il est alors clair que rajouter ou supprimer un arc revient à modifier un seul q_i : le calcul peut se faire de manière locale. Les méthodes de *recherche locales* sont alors utilisables.

Les scores utilisés pour l'apprentissage de réseau bayésien vérifient, en pratique, toutes ces propriétés.

Entropie conditionnelle et dimension d'un réseau bayésien

- L'entropie statistique, due à Claude Shannon, est une fonction mathématique qui correspond à la quantité d'information contenue ou délivrée par une source d'information. Elle est telle que plus la source est redondante, moins elle contient d'information au sens de Shannon.

Entropie conditionnelle dans un réseau bayésien

$$H(T, D) = \sum_{i=1}^{n} \sum_{j=1}^{q_i} \sum_{k=1}^{r_i} -\frac{N_{i,j,k}}{N_{i,j}} \cdot log_2 \left(\frac{N_{i,j,k}}{N_{i,j}} \right)$$

où r_i est le nombre de valeurs de X_i et $q_i = \prod_{j \in pa_i} r_j$ est le nombre de configuration des parents de X_i.

On peut prouver que $log_2 L(D : \Theta^{MV}, T) = -N \cdot H(T, D)$: **Maximiser la vraisemblance va avoir tendance à produire des réseaux bayésiens complet.**

- La dimension d'un réseau bayésien va être donnée par le nombre de paramètres nécessaires à l'instantiation de toutes les lois conditionnelles ($= |\Theta|$). En notant que pour la loi marginale d'une variable multinomiale X_i, il faut $r_i - 1$ paramètres, il est aisé de trouver que :

$$Dim(T) = \sum_i ((r_i - 1) \cdot q_i)$$

Quelques scores (1) : AIC/BIC

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

score AIC

- Akaike Information Criterion

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

score BIC

- Bayesian Information Criterion

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

Quelques scores (2) : MDL

MDL consiste à considérer la compacité de la représentation du modèle comme un bon critère de la qualité de ce modèle. Étonamment, ce critère est équivalent au critère BIC ci-dessus.
Il s'agit donc de minimiser la taille de la représentation, composée de :

- la représentation du modèle,
- la représentation des données sous forme de paramètres du modèle.

score MDL

- Minimum Description Length

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

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

Quelques scores (3) : BDe

Avec un critère bayésien, il s'agirait simplement de maximiser la probabilité jointe de T et D :

$$\begin{aligned} P(T, D) &= \int_{\Theta} P(D \mid \Theta, T) \cdot P(\Theta \mid T) \cdot P(T) d\Theta \\ &= P(T) \cdot \int_{\Theta} L(\Theta, T : D) \cdot P(\Theta \mid T) d\Theta \end{aligned}$$

Avec des hypothèses d'indépendances, un *a priori* de Dirichlet bien choisi, on obtient :

score BDe

- Bayesian Dirichlet score Equivalent

$$\text{Score}_{\text{BDe}}(T, D) = P(T) \prod_{i=1}^{n} \prod_{j=1}^{q_i} \frac{\Gamma(\alpha_{i,j})}{\Gamma(N_{i,j} + \alpha_{i,j})} \prod_{k=1}^{r_i} \frac{\Gamma(N_{i,j,k} + \alpha_{i,j,k})}{\Gamma(\alpha_{i,j,k})}$$

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 (les voisins d'un élément sont les points atteignables par l'application d'une opération élémentaire à cet élément).
- Soit un score (heurisitique) calculable localement.
- La recherche locale est alors une séquence de voisins tels qu'à partir du point initial, tout élément ultérieur de la séquence augmente le score. (*Greedy Search*).

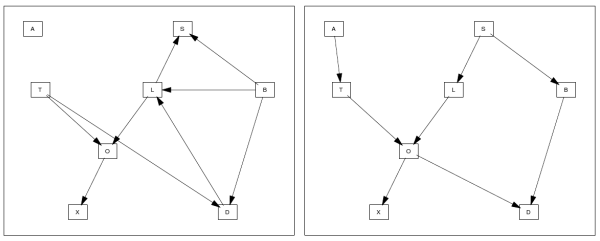
Recherche locale dans les réseaux bayésiens

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

Recherche locale : Greedy Search

Algorithme implémentant exactement ce qui est défini précédemment.

Réseau obtenu vs. théorique



L'algorithme peut converger vers des minima locaux.

Recherche locale : Diminution de l'espace de recherche

S'il existe un ordre dans les nœuds, tel qu'il ne soit pas possible d'avoir des arcs rétrogrades, alors il y a diminution de la taille de l'espace de recherche.

Taille de l'espace de recherche avec ordre sur les nœuds

$$NS'(n) = 2^{\frac{n\cdot(n-1)}{2}}$$

Algorithme K2

- Réseau initial sans arcs
- Opération élémentaire : ajouter un arc de j à $i > j$.
- Greedy algorithm sur le score BD(e).
- Limite sur le nombre de parents maximum.

Problème principal : algorithme très dépendant de l'ordre.

Changement d'espace : équivalents de Markov

Comme la recherche locale peut tomber dans des minima locaux, l'idée est de s'affranchir d'une partie de ces minima en changeant d'espace pour l'espace des classes d'équivalence de Markov.

Greedy Equivalence Search

L'espace des classes d'équivalences (notés les graphes essentiels) a une structure. On peut définir des opérateurs élémentaires et donc mener une recherche locale.

- Avantage : Pas de plage de score équivalence.
- Pas avantage : La taille de l'espace de recherche est sensiblement la même (ratio asymptotique de 3.7).