# *Sur l'apprentissage de Réseaux Bayésiens à partir de bases d'exemples incomplètes et application à la classification*

**Olivier FRANÇOIS** et Philippe LERAY,
Laboratoire **LITIS**, Rouen.

**Rencontres Inter-Associations**
La classification et ses applications
AFIA - ARIA - EGC - INFORSID - SFC - SFDS - LMO - ASTI
21 Mars 2006

## Plan



Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS

## Problématique

Soit $\mathcal{X}$ un système complexe.
$\mathcal{X}$ est représenté par de nombreux attributs $\{X_i\}_{1 \leqslant i \leqslant n}$.

- Certains attributs sont observés **systématiquement**,
- d'autres sont observés **occasionnellement**,
  - état *critique* du système ?
  - mesure *couteuse* ?...
- et de nombreux autres ne sont **jamais** observés,
  - parce que leur *influence/pertinence est faible* ?
  - parce que l'on ne *pas connaissance* de leur intérêt ?...

Par Exemple : Pour une base de 2000 exemples sur 20 attributs,
20% des mesures sont `manquantes complètement au hasard`
$\Longrightarrow$ en moyenne *seulement* 23 cas complets (c-à-d $\%EI \simeq 99\%$)

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS

## Types de donéees manquantes

Notations :

$$\mathbf{D} =< \mathbf{O}, \mathbf{H} >= ((d_{il}))_{n \times m}$$

$\mathbf{R} = ((r_{il}))_{n \times m}$, une matrice où $r_{il} = 1$ si $d_{il}$ est manquant, 0 sinon.
$\Theta$, paramètres de la loi qui a généré $\mathbf{D}$,
$\mu$, paramètres de la loi qui a généré $\mathbf{R}$.

Données manquantes ? Rubin (1976)

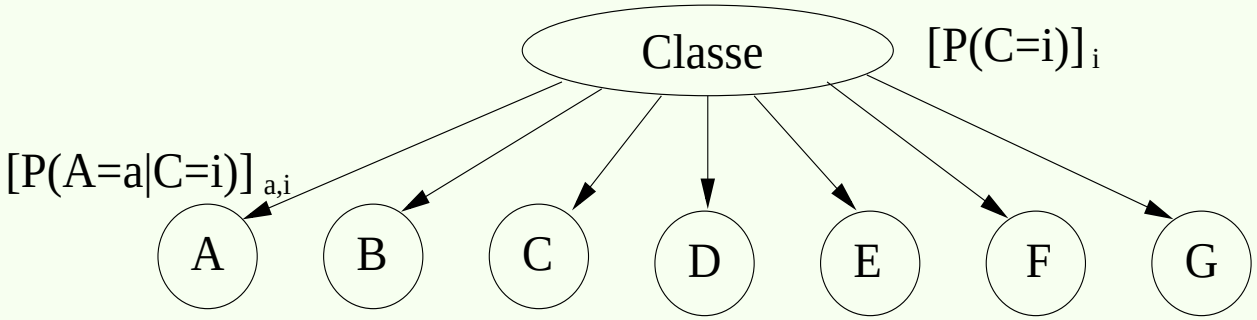
$$\mathbb{P}(\mathbf{O}, \mathbf{H}, \mathbf{R}|\Theta, \mu) = \mathbb{P}(\mathbf{O}, \mathbf{H}|\Theta) \times \mathbb{P}(\mathbf{R}|\mathbf{O}, \mathbf{H}, \mu)$$

- MCAR : $\mathbb{P}(\mathbf{R}|\mathbf{O}, \mathbf{H}, \mu) = \mathbb{P}(\mathbf{R}|\mu)$
- MAR : $\mathbb{P}(\mathbf{R}|\mathbf{O}, \mathbf{H}, \mu) = \mathbb{P}(\mathbf{R}|\mathbf{O}, \mu)$
- NMAR : $\mathbb{P}(\mathbf{R}|\mathbf{O}, \mathbf{H}, \mu)$, cas non ignorables.

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes LITIS

## *Le réseau Naïf*

Supposons que *la classe a une influence sur toutes les variables,* **mais** indépendamment

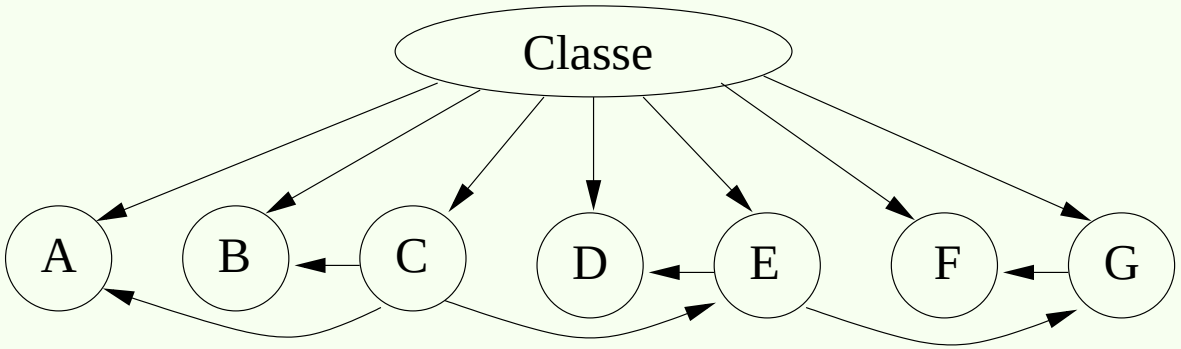



Ce qu'il est possible de faire :

- l'apprentissage des paramètres avec DI (par ex. avec EM),
- l'inférence avec des Données Incomplètes.

## *Le réseau Naïf*

Supposons que *la classe a une influence sur toutes les variables,* **mais** *plus* indépendamment




Ce qu'il est *toujours* possible de faire :

- l'apprentissage des paramètres avec DI (par ex. avec EM),
- l'inférence avec des Données Incomplètes.

et
si l'on veut ajouter des dépendances automatiquement : ?
(peu de méthodes efficaces à partir de DI)

## *Donner un score à partir d'une base incomplète*

Soit $\mathcal{S}(\mathcal{M}|\mathbf{D}_C)$, score pour un modèle $\mathcal{M}$ et des données complètes $\mathbf{D}_C$.

$\rightarrow$ approximation de $\mathcal{S}$ pour $\mathcal{M}$ et la base incomplète $\mathbf{D} =< \mathbf{O}, \mathbf{H} >$

$$\mathcal{Q}^{\mathcal{S}}(\mathcal{M}|\mathbf{D}) = \mathbb{E}_{\mathbf{H}\sim\mathbb{P}(\mathbf{H}|\mathbf{O},\mu)}\left[\mathcal{S}(\mathcal{M}|\mathbf{O},\mathbf{H})\right]$$

Mais la loi $\mathbb{P}(\mathbf{H}|\mathbf{O},\mu)$ est inconnue.

---

### Principe EM

Supposons que le modèle $\mathcal{M}^0$ a généré la base $\mathbf{D}$ alors

$$\mathcal{Q}^{\mathcal{S}}(\mathcal{M}|\mathbf{D}) \approx \mathcal{Q}^{\mathcal{S}}(\mathcal{M}:\mathcal{M}^0|\mathbf{D}) = \mathbb{E}_{\mathbf{H}\sim\mathbb{P}(\mathbf{H}|\mathbf{O},\mathcal{M}^0)}\left[\mathcal{S}(\mathcal{M}|\mathbf{O},\mathbf{H})\right]$$

$$\mathcal{Q}^{\mathcal{S}}(\mathcal{M}:\mathcal{M}^0|\mathbf{D}) = \sum_{\mathbf{H}} \mathcal{S}(\mathcal{M}|\mathbf{O},\mathbf{H})\mathbb{P}(\mathbf{H}|\mathbf{O},\mathcal{M}^0))$$

Ou la loi *a posteriori* $\mathbb{P}(\mathbf{H}|\mathbf{O},\mathcal{M}^0)$ est connue.

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes LITIS

## *Structural-EM*

- Choisir un modèle $\mathcal{M}^0$ ($\Rightarrow \mathbb{P}(\mathbf{H}|\mathcal{M}^0)$)
- Trouver un modèle $\mathcal{M}^{i+1}$ qui maximise** un score $Q^{\mathcal{S}}(\mathcal{M} : \mathcal{M}^i|\mathbf{D})$
- Utiliser le nouveau modèle comme base pour l'itération suivante jusqu'à convergence.

**EM généralisé : augmente le score

AMS-EM : le nouveau modèle est choisi parmis les *voisins* du graphe courant Friedman (1997)

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS

## *MWST-EM*

AMS-EM : le nouveau modèle est choisi parmis les *voisins* du graphe courant Friedman (1997) $\longrightarrow$ nombreuses itérations

**MWST-EM** : nous trouvons le 'meilleur' modèle dans l'espace des arbres Leray & François (2005)$\longrightarrow$ peu d'itérations

Pour cela, on utilise un algorithme de type `Kruskal` sur la matrice de score suivante :

$$\left[M_{ij}^{Q}\right]_{i,j} = \left[Q^{bic}(X_i, P_i = \{X_j\}, \Theta_{Xi|X_j} : \mathcal{T}^*, \Theta^*) - Q^{bic}(X_i, P_i = \emptyset, \Theta_{Xi} : \mathcal{T}^*, \Theta^*)\right]$$

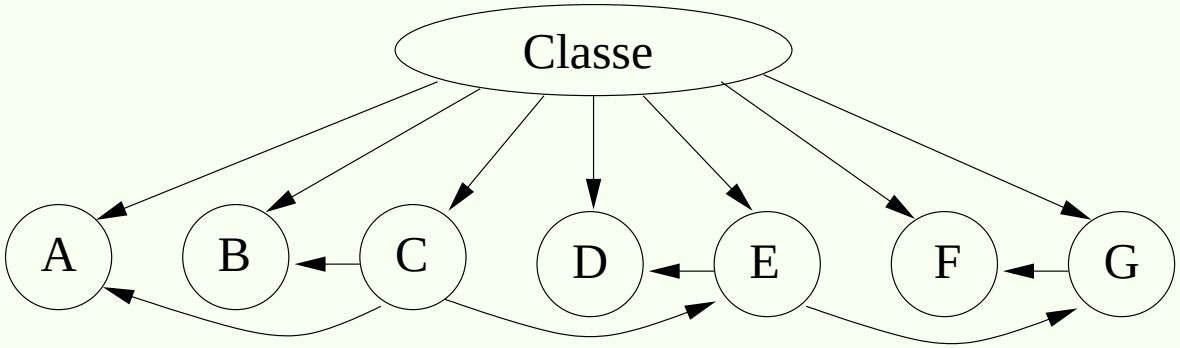
où $Q^{BIC}(\mathcal{G}, \Theta : \mathcal{G}^*, \Theta^*) = \sum_i Q^{bic}(X_i, P_i, \Theta_{Xi|P_i} : \mathcal{G}^*, \Theta^*)$ et

$$Q^{bic}(X_i, P_i, \Theta_{Xi|P_i} : \mathcal{G}^*, \Theta^*) = \sum_{X_i = x_k} \sum_{P_i = pa_j} N_{ijk}^* \log \theta_{ijk} - \frac{\log N}{2} \mathrm{Dim}(\Theta_{Xi|P_i})$$

avec $N_{ijk}^* = E_{\mathcal{G}^*, \Theta^*}\left[N_{ijk}\right] = N * P(X_i = x_k, P_i = pa_j | \mathcal{G}^*, \Theta^*)$.

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes LITIS

## TAN-EM




Pour augmenter le réseau naïf par un arbre,

la classe fait toujours partie de l'ensemble des parents

et la matrice de score devient ($i, j \neq classe$) :

$$\left[M_{ij}^{Q}\right]_{i,j} = \Big[Q^{bic}(X_i, P_i = \{C, X_j\}, \Theta_{Xi|X_j} : \mathcal{T}^*, \Theta^*) - Q^{bic}(X_i, P_i = \{C\}, \Theta_{Xi} : \mathcal{T}^*, \Theta^*)\Big]$$

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS

## Résultats

| | N | N app | N test | #C | %EI |
|---|---|---|---|---|---|
| Hepatitis | 20 | 90 | 65 | 2 | 8.4 |
| House | 17 | 290 | 145 | 2 | 46.7 |
| Horse | 28 | 300 | 300 | 2 | 88.0 |
| Thyroid | 22 | 2800 | 972 | 2 | 29.9 |
| Mushrooms | 23 | 5416 | 2708 | 2 | 30.5 |

| | NB-EM | MWST-EM | TAN-EM | AMS-EM | AMS-EM+T |
|---|---|---|---|---|---|
| Hepatitis | 70.8%<br>-1224 ; 29 | 73.8%<br>**-1147** ; 90 | **75.4**%<br>**-1148** ; 88 | 66.1%<br>-1211.5 ; 1213 | 66.1%<br>-1207 ; 1478 |
| House | 89.7%<br>-2203 ; 110 | **93.8**%<br>-2518 ; 157 | 92.4%<br>**-2022** ; 180 | 92.4%<br>-2524 ; 1732 | **93.8**%<br>-2195 ; 3327 |
| Horse | 75%<br>-5589 ; 227 | 77.9%<br>**-5199** ; 656 | **80.9**%<br>-5354 ; 582 | 66.2%<br>-5348 ; 31807 | 66.2%<br>-5318 ; 10054 |
| Thyroid | 95.3%<br>-39348 ; 1305 | 93.8%<br>-38881 ; 3173 | **96.2**%<br>**-38350** ; 3471 | 93.8%<br>**-38303** ; 17197 | 93.8%<br>-39749 ; 14492 |
| Mushrooms | **92.8**%<br>-97854 ; 2028 | 74.7%<br>-108011 ; 6228 | 91.3%<br>**-87556** ; 5987 | 74.9%<br>-111484 ; 70494 | 74.9%<br>-110828 ; 59795 |

## Conclusions

La méthode TAN-EM permet d'obtenir :

- de bonnes performances en classification,
- de bonnes vraissemblances des modèles obtenus,
- un excellent rapport *performances/rapidité*
  (car basée sur le réseau naïf et MWST-EM Leray & François (2005)).

Néanmoins, cette méthode

- est limitée au tâches de classification,
- augmente le NB *forcement* par un arbre.

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS

## Perspectives

Pour TAN-EM :

- adaptation à la classification non-supervisée,
- tests sur des données générées (MAR),
- tests en non-supervisé.

Pour MWST-EM :

- passer de l'espace des arbres à
  - l'espace des *forets* ($\Longrightarrow$ FAN-EM),
  - l'espace des *équivalents de Markov* ($\Longrightarrow$ GES-EM et BNAN-EM ?),
- remplacer les principes de EM par d'autres (Robust Bayesian Estimator...) $\Longrightarrow$ NMAR.

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS

Merci pour votre attention.

- Questions ?
- Remarques ?
- Suggestions ?

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS

**Friedman, N. (1997).**

**Learning belief networks in the presence of missing values and hidden variables.**

In *in the Proceedings of the 14th International Conference on Machine Learning*, (pp. 125–133). Morgan Kaufmann.

**Leray, P. & François, O. (2005).**

**Bayesian network structural learning and incomplete data.**

In *in the proccedings of the International and Interdisciplinary Conference on Adaptive Knowledge Representation and Reasoning (AKRR 2005), Espoo, Finland*.

**Rubin, D. (1976).**

**Inference and missing data.**

*Biometrika*, *63*, 581–592.

Laboratoire d'Informatique, de Traitement de l'Information et des Systèmes
LITIS