

Les arbres de Markov cachés

Hugo Gangloff

`hugo_gangloff@telecom-sudparis.eu`

Méthodes Statistiques pour la Segmentation d'Images
Octobre 2020



Plan

- 1** Introduction
- 2** Le modèle d'arbre de Markov de caché
- 3** Inférence dans les arbres de Markov cachés
- 4** Estimation des paramètres dans les arbres de Markov cachés
- 5** Arbres de Markov couples et triplets
- 6** Conclusion

Introduction : les modèles probabilistes graphiques orientés

Notions sur les graphes orientés

- Soit $\mathcal{G} = (\mathcal{S}, \mathcal{E})$ un graphe orienté
 - \mathcal{S} est l'ensemble des *sommets* (ou *sites* ou *nœuds*)
 - \mathcal{E} est l'ensemble des *arêtes orientées*

Notions sur les graphes orientés

- Soit $\mathcal{G} = (\mathcal{S}, \mathcal{E})$ un graphe orienté
 - \mathcal{S} est l'ensemble des *sommets* (ou *sites* ou *nœuds*)
 - \mathcal{E} est l'ensemble des *arêtes orientées*
- $\forall s \in \mathcal{S}$, s^- est le *père* de s , s'il existe, dans \mathcal{E} , une arête orientée de s^- vers s .
Dans ce cas également, s est le *fils* de s^- .

Notions sur les graphes orientés

- Soit $\mathcal{G} = (\mathcal{S}, \mathcal{E})$ un graphe orienté
 - \mathcal{S} est l'ensemble des *sommets* (ou *sites* ou *nœuds*)
 - \mathcal{E} est l'ensemble des *arêtes orientées*
- $\forall s \in \mathcal{S}$, s^- est le *père* de s , s'il existe, dans \mathcal{E} , une arête orientée de s^- vers s .
Dans ce cas également, s est le *fils* de s^- .
- On note $\mathcal{P}(s)$ l'ensemble des pères de s , $\forall s \in \mathcal{S}$.

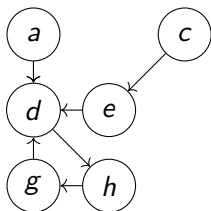
Notions sur les graphes orientés

- Soit $\mathcal{G} = (\mathcal{S}, \mathcal{E})$ un graphe orienté
 - \mathcal{S} est l'ensemble des *sommets* (ou *sites* ou *nœuds*)
 - \mathcal{E} est l'ensemble des *arêtes orientées*
- $\forall s \in \mathcal{S}$, s^- est le *père* de s , s'il existe, dans \mathcal{E} , une arête orientée de s^- vers s .
Dans ce cas également, s est le *fils* de s^- .
- On note $\mathcal{P}(s)$ l'ensemble des pères de s , $\forall s \in \mathcal{S}$.
- On appelle *racine* un sommet sans aucun père et $\mathcal{S}_{\mathcal{R}}$ est l'ensemble des racines de \mathcal{S} .

Notions sur les graphes orientés

- Soit $\mathcal{G} = (\mathcal{S}, \mathcal{E})$ un graphe orienté
 - \mathcal{S} est l'ensemble des *sommets* (ou *sites* ou *nœuds*)
 - \mathcal{E} est l'ensemble des *arêtes orientées*
- $\forall s \in \mathcal{S}$, s^- est le *père* de s , s'il existe, dans \mathcal{E} , une arête orientée de s^- vers s .
Dans ce cas également, s est le *fils* de s^- .
- On note $\mathcal{P}(s)$ l'ensemble des pères de s , $\forall s \in \mathcal{S}$.
- On appelle *racine* un sommet sans aucun père et $\mathcal{S}_{\mathcal{R}}$ est l'ensemble des racines de \mathcal{S} .
- On note $\bar{\mathcal{S}}$ l'ensemble des sommets qui ont au moins un père.

Notions sur les graphes orientés



Un graphe orienté

$$\mathcal{S} = \{a, c, d, e, g, h\}$$

$$\bar{\mathcal{S}} = \{d, e, g, h\}$$

a et c sont des racines

a est le père de d

d est le fils de a

$$\mathcal{P}(d) = \{a, e, g\}$$

$\{d, g, h\}$ est un cycle orienté

Exemple

Les modèles probabilistes graphiques orientés

- On associe une variable aléatoire X_s à chaque $s \in \mathcal{S}$.

Les modèles probabilistes graphiques orientés

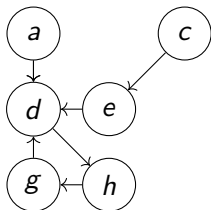
- On associe une variable aléatoire X_s à chaque $s \in \mathcal{S}$.
- Un modèle probabiliste graphique orienté est défini à partir de lois conditionnelles locales $p(X_s | \mathbf{X}_{\mathcal{P}(s)}), \forall s \in \bar{\mathcal{S}}$ et $p(X_r), \forall r \in \mathcal{S}_{\mathcal{R}}$.

Les modèles probabilistes graphiques orientés

- On associe une variable aléatoire X_s à chaque $s \in \mathcal{S}$.
- Un modèle probabiliste graphique orienté est défini à partir de lois conditionnelles locales $p(X_s | \mathbf{X}_{\mathcal{P}(s)}), \forall s \in \bar{\mathcal{S}}$ et $p(X_r), \forall r \in \mathcal{S}_{\mathcal{R}}$.
- En appliquant la définition du conditionnement, la distribution de probabilité du vecteur aléatoire $\mathbf{X} = \{X_s\}_{s \in \mathcal{S}}$ sur \mathcal{G} s'écrit:

$$p(\mathbf{X}) = \prod_{r \in \mathcal{S}_{\mathcal{R}}} p(X_r) \prod_{s \in \bar{\mathcal{S}}} p(X_s | \mathbf{X}_{\mathcal{P}(s)})$$

Les modèles probabilistes graphiques orientés

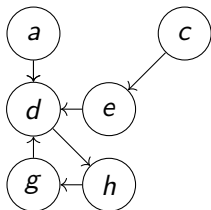


Un graphe orienté

$$\begin{aligned} p(X_a, X_b, X_c, X_d, X_e, X_f, X_h, X_g) = \\ p(X_a)p(X_c) \times \\ p(X_e|X_c)p(X_d|X_a, X_e, X_g) \times \\ p(X_h|X_d)p(X_g|X_h) \end{aligned}$$

Exemple

Les modèles probabilistes graphiques orientés



Un graphe orienté

$$\begin{aligned} p(X_a, X_b, X_c, X_d, X_e, X_f, X_h, X_g) = \\ p(X_a)p(X_c) \times \\ p(X_e|X_c)p(X_d|X_a, X_e, X_g) \times \\ p(X_h|X_d)p(X_g|X_h) \end{aligned}$$

Exemple

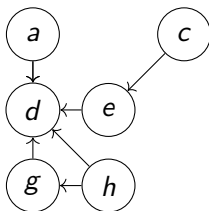
→ On va restreindre ce cadre très général où l'inférence est complexe, pour arriver aux arbres de Markov et leurs propriétés très utiles!

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...



$\{d, g, h\}$ est un
semi-cycle.

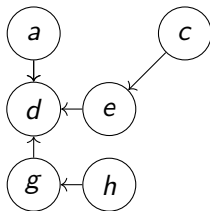
On obtient les **graphes orientés acycliques** (ou réseaux bayésiens)

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...



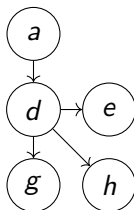
On obtient les *arbres orientés*

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...
- et limitation à un seul nœud racine...

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...
- et limitation à un seul nœud racine...



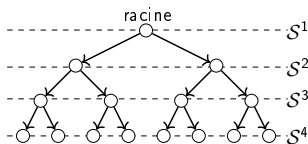
On obtient les *arborescences*

Restrictions et hypothèses sur les graphes orientés

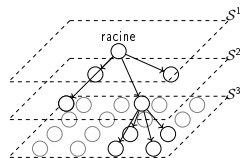
- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...
- et limitation à un seul nœud racine...
- et organisation en résolutions (ou générations) :
 - chaque nœud a 2 ou 4 fils (sauf sur la dernière résolution)

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...
- et limitation à un seul nœud racine...
- et organisation en résolutions (ou générations) :
 - chaque nœud a 2 ou 4 fils (sauf sur la dernière résolution)



diarbre de Markov



quadarbre de Markov

*On obtient les **arbres de Markov***

Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...
- et limitation à un seul nœud racine...
- et organisation en résolutions (ou générations) :
 - chaque nœud a 2 ou 4 fils (sauf sur la dernière résolution)

*On obtient les **arbres de Markov***

- **Cas particulier** : chaque nœud a 1 fils (sauf sur la dernière résolution)



Restrictions et hypothèses sur les graphes orientés

- Interdiction des cycles orientés...
- et interdiction des semi cycles orientés...
- et limitation à un seul nœud racine...
- et organisation en résolutions (ou générations) :
 - chaque nœud a 2 ou 4 fils (sauf sur la dernière résolution)

Le modèle d'arbre de Markov de caché

Le contexte bayésien de la segmentation d'images

On observe une image à valeurs réelles et nous souhaitons la segmenter en K classes $\{\omega_k\}_{k \in \{1, \dots, K\}}$ ($\Omega \triangleq \{\omega_k\}_{k \in \{1, \dots, K\}}$).

- En chacun des sites s de \mathcal{G} on associe un couple de variables aléatoires (X_s, Y_s) .

Le contexte bayésien de la segmentation d'images

On observe une image à valeurs réelles et nous souhaitons la segmenter en K classes $\{\omega_k\}_{k \in \{1, \dots, K\}}$ ($\Omega \triangleq \{\omega_k\}_{k \in \{1, \dots, K\}}$).

- En chacun des sites s de \mathcal{G} on associe un couple de variables aléatoires (X_s, Y_s) .
- Le vecteur aléatoire $\mathbf{X} = (X_s)_{s \in \mathcal{S}}$ à valeurs dans $\Omega^{|\mathcal{S}|}$ est composé des **variables cachées**.

Le contexte bayésien de la segmentation d'images

On observe une image à valeurs réelles et nous souhaitons la segmenter en K classes $\{\omega_k\}_{k \in \{1, \dots, K\}}$ ($\Omega \triangleq \{\omega_k\}_{k \in \{1, \dots, K\}}$).

- En chacun des sites s de \mathcal{G} on associe un couple de variables aléatoires (X_s, Y_s) .
- Le vecteur aléatoire $\mathbf{X} = (X_s)_{s \in \mathcal{S}}$ à valeurs dans $\Omega^{|\mathcal{S}|}$ est composé des **variables cachées**.
- Le vecteur aléatoire $\mathbf{Y} = (Y_s)_{s \in \mathcal{S}}$ à valeurs dans $\mathbb{R}^{|\mathcal{S}|}$ est composé des **variables observées**.

Le contexte bayésien de la segmentation d'images

- Nous voulons calculer **la loi *a posteriori*** $p(\mathbf{X}|\mathbf{Y})$ des variables cachées sachant les variables observées.

Grâce au théorème de Bayes on a :

$$\begin{aligned} p(\mathbf{X}|\mathbf{Y}) &= \frac{p(\mathbf{X}, \mathbf{Y})}{p(\mathbf{Y})}, \\ &= \frac{p(\mathbf{X})p(\mathbf{Y}|\mathbf{X})}{p(\mathbf{Y})}, \end{aligned}$$

Dans notre cas où l'observation est constante :

$$p(\mathbf{X}|\mathbf{Y}) \propto p(\mathbf{X})p(\mathbf{Y}|\mathbf{X}).$$

Le contexte bayésien de la segmentation d'images

- Nous voulons calculer la loi *a posteriori* $p(\mathbf{X}|\mathbf{Y})$ des variables cachées sachant les variables observées.

Grâce au théorème de Bayes on a :

$$\begin{aligned} p(\mathbf{X}|\mathbf{Y}) &= \frac{p(\mathbf{X}, \mathbf{Y})}{p(\mathbf{Y})}, \\ &= \frac{p(\mathbf{X})p(\mathbf{Y}|\mathbf{X})}{p(\mathbf{Y})}, \end{aligned}$$

Dans notre cas où l'observation est constante :

$$p(\mathbf{X}|\mathbf{Y}) \propto p(\mathbf{X})p(\mathbf{Y}|\mathbf{X}).$$

- Ainsi la loi *a posteriori* peut être calculée si l'on a :
 - $p(\mathbf{X})$, la loi *a priori*.
 - $p(\mathbf{Y}|\mathbf{X})$, la *vraisemblance conditionnelle*.
 - ou, de manière équivalente, la loi jointe $p(\mathbf{X}, \mathbf{Y})$.

Le contexte bayésien de la segmentation d'images

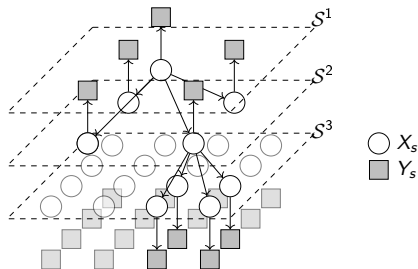
- S'il nous n'avons pas les paramètres des distributions mises en jeu, il nous faut les estimer : c'est l'étape d'**estimation des paramètres**.

Le contexte bayésien de la segmentation d'images

- S'il nous n'avons pas les paramètres des distributions mises en jeu, il nous faut les estimer : c'est l'étape d'**estimation des paramètres**.
- Lorsque la loi *a posteriori* est calculée, il reste à estimer les états cachés selon un critère donné : c'est l'étape d'**inférence**.

Le modèle d'arbre de Markov caché I

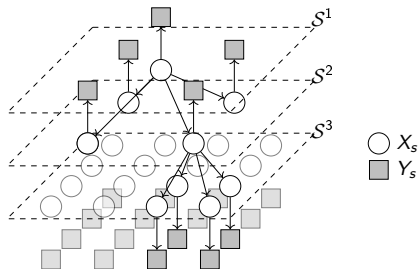
On associe, $\forall s \in \mathcal{S}$, un couple (X_s, Y_s) à chaque sommet du graphe d'arbre de Markov établi précédemment :



On obtient la représentation graphique d'un **arbre de Markov caché** (quadarbre à 3 résolutions à bruit indépendant ici).

Le modèle d'arbre de Markov caché I

On associe, $\forall s \in \mathcal{S}$, un couple (X_s, Y_s) à chaque sommet du graphe d'arbre de Markov établi précédemment :



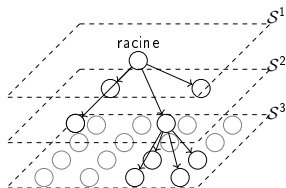
→ Écrivons la loi
de probabilité
 $p(\mathbf{X}, \mathbf{Y})$ associée à
ce modèle
graphique orienté

On obtient la représentation graphique d'un **arbre de Markov caché** (quadarbre à 3 résolutions à bruit indépendant ici).

Le modèle d'arbre de Markov caché II

Loi *a priori* : loi d'un arbre de Markov

$$p(\mathbf{X}) =$$

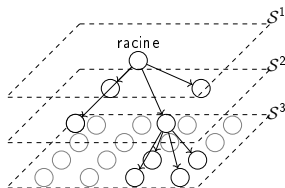


Le modèle d'arbre de Markov caché II

Loi *a priori* : loi d'un arbre de Markov

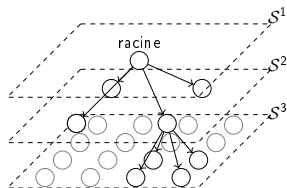
$$p(\mathbf{X}) = p(X_r) \prod_{n \in \{2, \dots, N\}} p(\mathbf{X}_{S^n} | \mathbf{X}_{S^{n-1}}),$$

H1 : chaîne de Markov en résolutions



Le modèle d'arbre de Markov caché II

Loi *a priori* : loi d'un arbre de Markov



$$p(\mathbf{X}) = p(X_r) \underbrace{\prod_{n \in \{2, \dots, N\}} p(\mathbf{X}_{S^n} | \mathbf{X}_{S^{n-1}})}_{\text{H1 : chaîne de Markov en résolutions}},$$

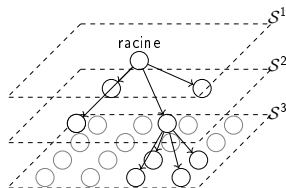
H1 : chaîne de Markov en résolutions

$$= p(X_r) \underbrace{\prod_{n \in \{1, \dots, N\}} \prod_{s \in S^n} p(X_s | \mathbf{X}_{S^{n-1}})}_{\text{H2 : variables d'une résolution indépendantes conditionnellement à la résolution précédente}},$$

H2 : variables d'une résolution indépendantes conditionnellement à la résolution précédente

Le modèle d'arbre de Markov caché II

Loi *a priori* : loi d'un arbre de Markov



$$p(\mathbf{X}) = p(X_r) \underbrace{\prod_{n \in \{2, \dots, N\}} p(\mathbf{X}_{S^n} | \mathbf{X}_{S^{n-1}})}_{\text{H1 : chaîne de Markov en résolutions}},$$

H1 : chaîne de Markov en résolutions

$$= p(X_r) \underbrace{\prod_{n \in \{1, \dots, N\}} \prod_{s \in S^n} p(X_s | \mathbf{X}_{S^{n-1}})}_{\text{H2 : variables d'une résolution indépendantes conditionnellement à la résolution précédente}},$$

H2 : variables d'une résolution indépendantes conditionnellement à la résolution précédente

$$= p(X_r) \underbrace{\prod_{s \in \bar{S}} p(X_s | X_{s-})}_{\text{H3 : } p(X_s | \mathbf{X}_{S^{n-1}}) = p(X_s | X_{s-})},$$

H3 : $p(X_s | \mathbf{X}_{S^{n-1}}) = p(X_s | X_{s-})$

Le modèle d'arbre de Markov caché II

Vraisemblance conditionnelle

Les hypothèses classiques d'un **bruit indépendant** sont :

$$p(\mathbf{Y}|\mathbf{X} = \mathbf{x}) =$$

Le modèle d'arbre de Markov caché II

Vraisemblance conditionnelle

Les hypothèses classiques d'un **bruit indépendant** sont :

$$p(\mathbf{Y}|\mathbf{X} = \mathbf{x}) = \underbrace{\prod_{s \in \mathcal{S}} p(Y_s|\mathbf{X} = \mathbf{x})}_{\text{H1}},$$

- H1 : les $(Y_s)_{s \in \mathcal{S}}$ sont indépendantes conditionnellement à $\mathbf{X} = \mathbf{x}$

Le modèle d'arbre de Markov caché II

Vraisemblance conditionnelle

Les hypothèses classiques d'un **bruit indépendant** sont :

$$\begin{aligned} p(\mathbf{Y}|\mathbf{X} = \mathbf{x}) &= \underbrace{\prod_{s \in \mathcal{S}} p(Y_s|\mathbf{X} = \mathbf{x})}_{\text{H1}} \\ &= \underbrace{\prod_{s \in \mathcal{S}} p(Y_s|X_s = x_s)}_{\text{H2}}. \end{aligned}$$

- H1 : les $(Y_s)_{s \in \mathcal{S}}$ sont indépendantes conditionnellement à $\mathbf{X} = \mathbf{x}$
- H2 : la loi de chaque Y_s conditionnellement à $\mathbf{X} = \mathbf{x}$ se réduit à un conditionnement par rapport à $X_s = x_s$.

Le modèle d'arbre de Markov caché II

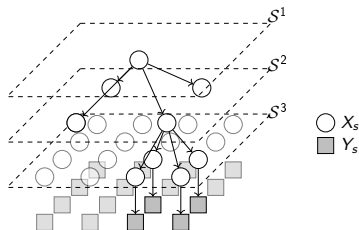
La distribution d'un arbre de Markov caché s'écrit donc:

$$p(\mathbf{X}, \mathbf{Y}) = \underbrace{p(X_r) \prod_{s \in \bar{\mathcal{S}}} p(X_s | X_{s-})}_{\substack{\text{loi d'un arbre de Markov} \\ \rightarrow \text{a priori}}} \underbrace{\prod_{s \in \mathcal{S}} p(Y_s | X_s)}_{\substack{\text{bruit indépendant} \\ \rightarrow \text{vraisemblance cond.}}} .$$

Le modèle d'arbre de Markov caché III

Nous supposons dans la suite que :

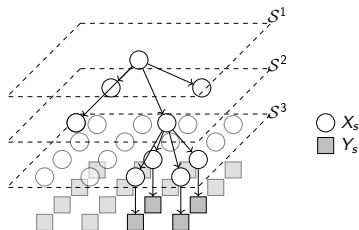
- Des observations sont seulement associées à la dernière résolution \mathcal{S}^N .
- Les fonctions de vraisemblance sont des gaussiennes dont les paramètres dépendent des états cachés sous-jacents.



Le modèle d'arbre de Markov caché III

Nous supposons dans la suite que :

- Des observations sont seulement associées à la dernière résolution \mathcal{S}^N .
- Les fonctions de vraisemblance sont des gaussiennes dont les paramètres dépendent des états cachés sous-jacents.

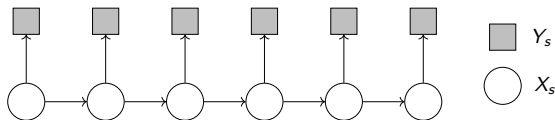


La distribution s'écrit dans ce cas:

$$p(\mathbf{X}, \mathbf{Y}) = p(X_r) \prod_{s \in \bar{\mathcal{S}}} p(X_s | X_{s-}) \prod_{s^N \in \mathcal{S}^N} p(Y_{s^N} | X_{s^N}).$$

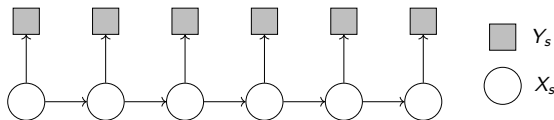
Le modèle d'arbre de Markov caché III

Une chaîne de Markov est un cas particulier : chaque sommet s , $\forall s \in \mathcal{S} \setminus \mathcal{S}^N$, a un seul fils.



Le modèle d'arbre de Markov caché III

Une chaîne de Markov est un cas particulier : chaque sommet s , $\forall s \in \mathcal{S} \setminus \mathcal{S}^N$, a un seul fils.



La distribution s'écrit:

$$p(\mathbf{X}, \mathbf{Y}) = p(X_r)p(Y_r|X_r) \prod_{s \in \bar{\mathcal{S}}} p(X_s|X_{s-})p(Y_s|X_s).$$

Simulations d'arbres de Markov

On oublie le processus \mathbf{Y} pour le moment

Exemple 1

Dans le cas de 2 classes ω_1 et ω_2 , avec:

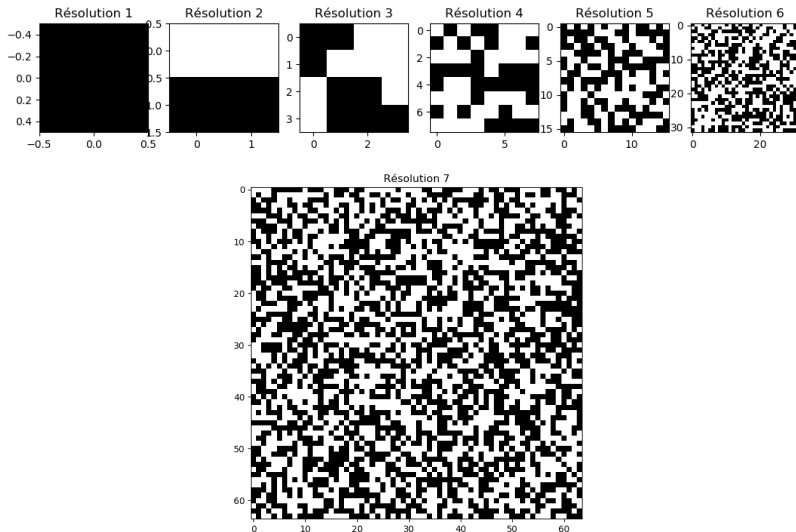
$$p(X_r = \omega_1) = p(X_r = \omega_2) = 0.5 \text{ et}$$

$$\begin{cases} p(X_s = \omega_1 | X_{s-} = \omega_1) = p(X_s = \omega_2 | X_{s-} = \omega_2) = 0.5, \\ p(X_s = \omega_1 | X_{s-} = \omega_2) = p(X_s = \omega_2 | X_{s-} = \omega_1) = 0.5. \end{cases}$$

→ La réalisation du père n'a pas d'incidence sur la réalisation du fils

Simulations d'arbres de Markov

Exemple 1



Simulations d'arbres de Markov

On oublie le processus \mathbf{Y} pour le moment

Exemple 2

Dans le cas de 2 classes ω_1 et ω_2 , avec:

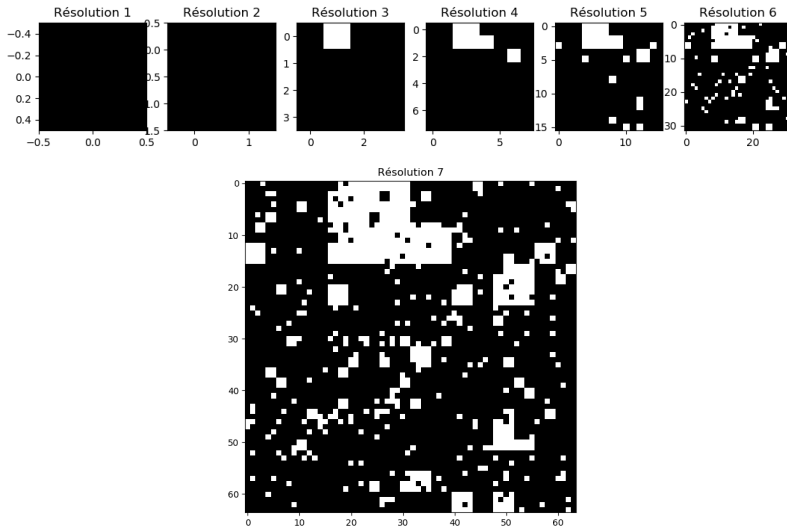
$$p(X_r = \omega_1) = p(X_r = \omega_2) = 0.5 \text{ et}$$

$$\begin{cases} p(X_s = \omega_1 | X_{s-} = \omega_1) = p(X_s = \omega_2 | X_{s-} = \omega_2) = 0.95, \\ p(X_s = \omega_1 | X_{s-} = \omega_2) = p(X_s = \omega_2 | X_{s-} = \omega_1) = 0.05. \end{cases}$$

→ On favorise les réalisations des fils de même classe que les réalisations des pères

Simulations d'arbres de Markov

Exemple 2



Simulations d'arbres de Markov

On oublie le processus \mathbf{Y} pour le moment

Exemple 3

Dans le cas de 2 classes ω_1 et ω_2 , avec:

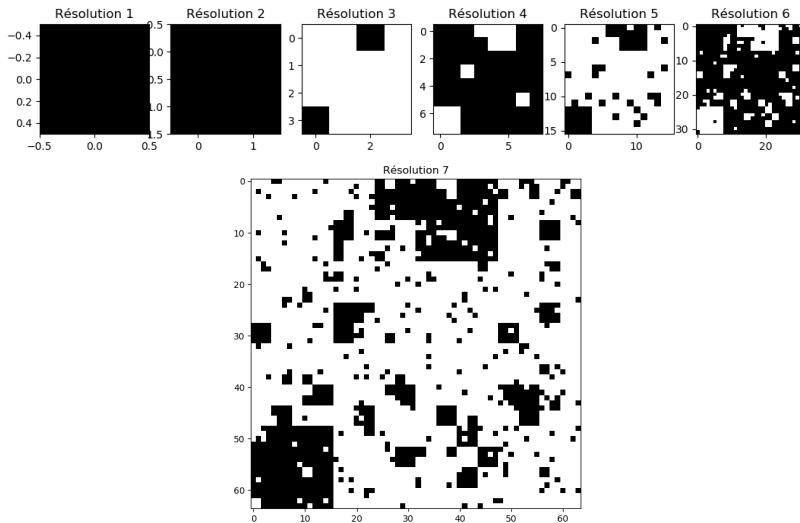
$$p(X_r = \omega_1) = p(X_r = \omega_2) = 0.5 \text{ et}$$

$$\begin{cases} p(X_s = \omega_1 | X_{s-} = \omega_1) = p(X_s = \omega_2 | X_{s-} = \omega_2) = 0.05, \\ p(X_s = \omega_1 | X_{s-} = \omega_2) = p(X_s = \omega_2 | X_{s-} = \omega_1) = 0.95. \end{cases}$$

→ On favorise les réalisations des fils de classe différente que les réalisations des pères

Simulations d'arbres de Markov

Exemple 3



Simulations d'arbres de Markov

Remarques :

- On remarque l'effet de "bloc" caractéristique des arbres de Markov, particulièrement pour les quadarbres.
- Les diarbres ont également été appliqués aux images avec d'autres "effets" caractéristiques.
- Les transitions ne sont pas nécessairement stationnaires, elles peuvent varier selon la résolution. C'est exploré dans le modèle **d'arbres de Markov évolutifs**.

Inférence dans les arbres de Markov cachés

Les quantités β

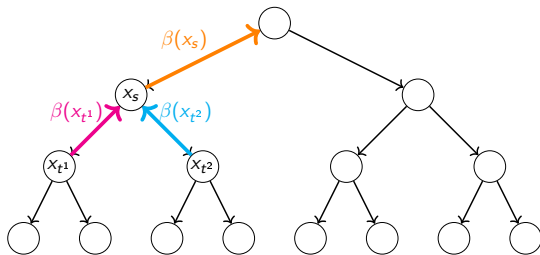
Soit \mathbf{s}^+ l'ensemble des fils de s . On calcule les quantités β ,
 $\forall x_s \in \Omega, \forall s \in \mathcal{S}^*/$

$$\begin{cases} \beta(x_s) = p(y_s|x_s), & \text{si } s \in \mathcal{S}^N, \\ \beta(x_s) = \prod_{t \in \mathbf{s}^+} (\sum_{x_t} \beta(x_t)p(x_t|x_s)) & \text{sinon.} \end{cases}$$

Il s'agit d'une **passee montante** (*upward*).

Les quantités β

Exemple de calcul dans un diarbre:



$$\beta(x_s) = \sum_{x_{t1}} \beta(x_{t1}) p(x_{t1} | x_s) \times \sum_{x_{t2}} \beta(x_{t2}) p(x_{t2} | x_s)$$

Loi de \mathbf{X} a posteriori

À partir des quantités β on calcule facilement les transitions a posteriori:

■ $\forall (x_s, x_{s-}) \in \Omega^2, \forall s \in \bar{\mathcal{S}}:$

$$p(x_s | x_{s-}, \mathbf{y}) = \frac{\beta(x_s) p(x_s | x_{s-})}{\sum_{x_s} \beta(x_s) p(x_s | x_{s-})},$$

■ $\forall x_r \in \Omega:$

$$p(x_r | \mathbf{y}) = \frac{\beta(x_r) p(x_r)}{\sum_{x_r} \beta(x_r) p(x_r)}.$$

→ La loi de \mathbf{X} a posteriori est celle d'un arbre de Markov

Maximum Posterior Mode (MPM)

Critère du MPM:

$$\forall s \in \mathcal{S}, \hat{x}_s^{MPM} = \operatorname{argmax}_{x_s} p(x_s | \mathbf{y}).$$

Maximum Posterior Mode (MPM)

Critère du MPM:

$$\forall s \in \mathcal{S}, \hat{x}_s^{MPM} = \operatorname{argmax}_{x_s} p(x_s | \mathbf{y}).$$

On peut calculer les transitions *a posteriori* en chacun des sites grâce aux quantités β : **pas**se montante (*upward*).

Maximum Posterior Mode (MPM)

Critère du MPM:

$$\forall s \in \mathcal{S}, \hat{x}_s^{MPM} = \operatorname{argmax}_{x_s} p(x_s | \mathbf{y}).$$

On peut calculer les transitions *a posteriori* en chacun des sites grâce aux quantités β : **passee montante** (*upward*).

Puis on effectue une **passee descendante** (*downward*):

■ $\forall x_r \in \Omega$:

$$p(x_r | \mathbf{y}) = \frac{\beta(x_r) p(x_r)}{\sum_{x_r} \beta(x_r) p(x_r)}.$$

■ $\forall x_s \in \Omega, \forall s \in \bar{\mathcal{S}}$:

$$p(x_s | \mathbf{y}) = \sum_{x_{s-}} p(x_{s-} | \mathbf{y}) p(x_s | x_{s-}, \mathbf{y}).$$

Maximum Posterior Mode (MPM)

Critère du MPM:

$$\forall s \in \mathcal{S}, \hat{x}_s^{MPM} = \operatorname{argmax}_{x_s} p(x_s | \mathbf{y}).$$

On peut calculer les transitions *a posteriori* en chacun des sites grâce aux quantités β : **passee montante** (*upward*).

Puis on effectue une **passee descendante** (*downward*):

■ $\forall x_r \in \Omega$:

$$p(x_r | \mathbf{y}) = \frac{\beta(x_r) p(x_r)}{\sum_{x_r} \beta(x_r) p(x_r)}.$$

■ $\forall x_s \in \Omega, \forall s \in \bar{\mathcal{S}}$:

$$p(x_s | \mathbf{y}) = \sum_{x_{s-}} p(x_{s-} | \mathbf{y}) p(x_s | x_{s-}, \mathbf{y}).$$

N.B. : Cet algorithme est aussi appelé *upward/downward*.

Maximum A Posteriori (MAP)

Critère du MAP:

$$\hat{\mathbf{x}}^{MAP} = \operatorname{argmax}_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}).$$

Maximum A Posteriori (MAP)

Critère du MAP:

$$\hat{\mathbf{x}}^{MAP} = \operatorname{argmax}_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}).$$

Une **passee montante**, $\forall x_{s-} \in \Omega, \forall s \in \bar{\mathcal{S}}$:

$$\begin{cases} \begin{cases} \nu_s(x_{s-}) = \max_{x_s} p(y_s|x_s)p(x_s|x_{s-}), \\ x_s^*(x_{s-}) = \operatorname{argmax}_{x_s} p(y_s|x_s)p(x_s|x_{s-}), \end{cases} & \text{si } s \in \mathcal{S}^N, \\ \begin{cases} \nu_s(x_{s-}) = \max_{x_s} p(x_s|x_{s-}) \prod_{t \in \mathbf{s}^+} \nu_t(x_s), \\ x_s^*(x_{s-}) = \operatorname{argmax}_{x_s} p(x_s|x_{s-}) \prod_{t \in \mathbf{s}^+} \nu_t(x_s), \end{cases} & \text{sinon.} \end{cases}$$

Maximum A Posteriori (MAP)

Critère du MAP:

$$\hat{\mathbf{x}}^{MAP} = \operatorname{argmax}_{\mathbf{x}} p(\mathbf{x}|\mathbf{y}).$$

Une **passee montante**, $\forall x_{s-} \in \Omega, \forall s \in \bar{\mathcal{S}}$:

$$\begin{cases} \begin{cases} \nu_s(x_{s-}) = \max_{x_s} p(y_s|x_s)p(x_s|x_{s-}), \\ x_s^*(x_{s-}) = \operatorname{argmax}_{x_s} p(y_s|x_s)p(x_s|x_{s-}), \end{cases} & \text{si } s \in \mathcal{S}^N, \\ \begin{cases} \nu_s(x_{s-}) = \max_{x_s} p(x_s|x_{s-}) \prod_{t \in \mathcal{S}^+} \nu_t(x_s), \\ x_s^*(x_{s-}) = \operatorname{argmax}_{x_s} p(x_s|x_{s-}) \prod_{t \in \mathcal{S}^+} \nu_t(x_s), \end{cases} & \text{sinon.} \end{cases}$$

Une **passee descendante**:

$$\begin{cases} \hat{x}_r^{MAP} = \operatorname{argmax}_{x_r} p(x_r) \prod_{s \in \mathcal{R}^+} \nu_s(x_r), \text{ } r \text{ racine de l'arbre,} \\ \hat{x}_s^{MAP} = x_s^*(\hat{x}_{s-}^{MAP}), \forall s \in \bar{\mathcal{S}}. \end{cases}$$

Remarques I

Liens avec les chaînes de Markov cachées :

- Nous venons de voir deux algorithmes d'inférence sous leur forme spécifique aux arbres de Markov cachés. Nous pouvons remarquer que:

Remarques I

Liens avec les chaînes de Markov cachées :

- Nous venons de voir deux algorithmes d'inférence sous leur forme spécifique aux arbres de Markov cachés. Nous pouvons remarquer que:
 - L'algorithme de calcul du MAP est une généralisation de **l'algorithme de Viterbi** pour le calcul du MAP dans les chaîne de Markov cachées.

Remarques I

Liens avec les chaînes de Markov cachées :

- Nous venons de voir deux algorithmes d'inférence sous leur forme spécifique aux arbres de Markov cachés. Nous pouvons remarquer que:
 - L'algorithme de calcul du MAP est une généralisation de **l'algorithme de Viterbi** pour le calcul du MAP dans les chaîne de Markov cachées.
 - L'algorithme *upward/downward* de calcul du MPM est une généralisation de **l'algorithme *forward/backward*** pour le calcul du MPM dans les chaînes de Markov cachées.

Remarques I

Liens avec les chaînes de Markov cachées :

- Nous venons de voir deux algorithmes d'inférence sous leur forme spécifique aux arbres de Markov cachés. Nous pouvons remarquer que:
 - L'algorithme de calcul du MAP est une généralisation de **l'algorithme de Viterbi** pour le calcul du MAP dans les chaîne de Markov cachées.
 - L'algorithme *upward/downward* de calcul du MPM est une généralisation de **l'algorithme *forward/backward*** pour le calcul du MPM dans les chaînes de Markov cachées.
- Pour les arbres comme pour les chaînes : l'inférence donne lieu à des procédures **déterministes et non itératives**.

Remarques II

Liens avec notre introduction :

- De plus, les algorithmes présentés peuvent être **réinterprétés** dans le cadre très large des algorithmes à propagation de messages que l'on retrouve dans la théorie générale des modèles graphiques probabilistes.

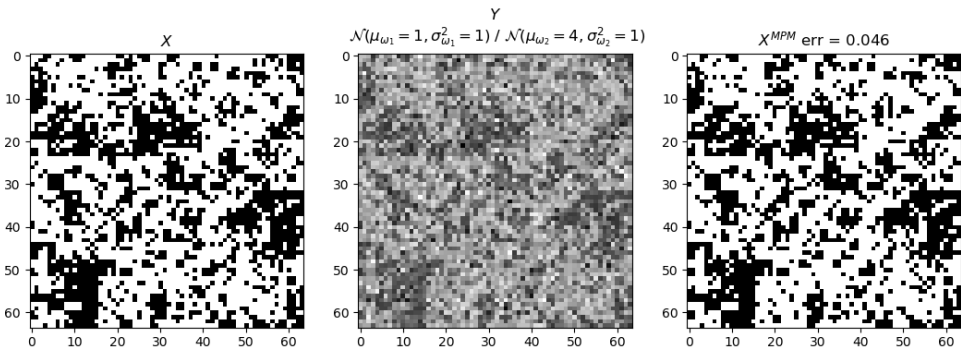
Remarques II

Liens avec notre introduction :

- De plus, les algorithmes présentés peuvent être **réinterprétés** dans le cadre très large des algorithmes à propagation de messages que l'on retrouve dans la théorie générale des modèles graphiques probabilistes.
- Le chapitre 20 du livre *Machine Learning, a probabilistic perspective* de Kevin Murphy (MIT Press, 2012) est une bonne introduction sur le sujet.

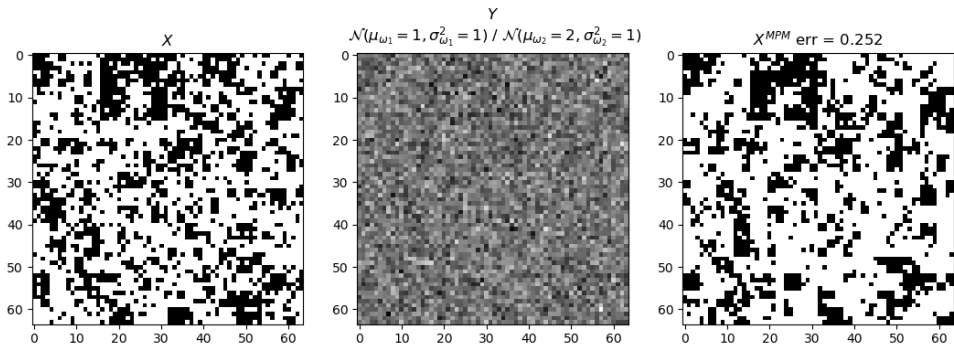
Segmentation supervisée

Ex. 1 : arbre simulé et bruit gaussien indépendant "faible"



Segmentation supervisée

Ex. 2 : arbre simulé et bruit gaussien indépendant "fort"



Estimation des paramètres dans les arbres de Markov cachés

Expectation Maximization (EM)

La loi jointe d'un arbre de Markov s'écrit:

$$p(\mathbf{x}, \mathbf{y}) = p(x_r) \prod_{s \in \bar{\mathcal{S}}} p(x_s | x_{s-}) \prod_{s^N \in \mathcal{S}^N} p(y_s^N | x_s^N).$$

Expectation Maximization (EM)

La loi jointe d'un arbre de Markov s'écrit:

$$p(\mathbf{x}, \mathbf{y}) = p(x_r) \prod_{s \in \bar{\mathcal{S}}} p(x_s | x_{s-}) \prod_{s^N \in \mathcal{S}^N} p(y_s^N | x_s^N).$$

Dans EM nous souhaitons maximiser:

$$Q(\boldsymbol{\theta} | \boldsymbol{\theta}^{(t)}) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x} | \mathbf{Y} = \mathbf{y}, \boldsymbol{\theta}^{(t)})} [\log p(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta})].$$

Expectation Maximization (EM)

La loi jointe d'un arbre de Markov s'écrit:

$$p(\mathbf{x}, \mathbf{y}) = p(x_r) \prod_{s \in \bar{S}} p(x_s | x_{s-}) \prod_{s^N \in S^N} p(y_s^N | x_s^N).$$

Dans EM nous souhaitons maximiser:

$$Q(\boldsymbol{\theta} | \boldsymbol{\theta}^{(t)}) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{X} | \mathbf{Y} = \mathbf{y}, \boldsymbol{\theta}^{(t)})} [\log p(\mathbf{x}, \mathbf{y}; \boldsymbol{\theta})].$$

Posons
$$\begin{cases} \pi_{\omega_i} = p(x_r = \omega_i), \forall \omega_i \in \Omega, \\ a_{\omega_i, \omega_j} = p(x_s = \omega_j | x_{s-} = \omega_i), \forall (\omega_i, \omega_j) \in \Omega^2. \end{cases}$$

Dans le cas du bruit gaussien indépendant, dont les densités f sont paramétrées par les états cachés :

$$\boldsymbol{\theta} = \{ \{ \pi_{\omega_i} \}, \{ a_{\omega_i, \omega_j} \}, \{ \mu_{\omega_i} \}, \{ \sigma_{\omega_i} \} \}.$$

Expectation Maximization (EM)

$$\boldsymbol{\theta} = \left\{ \{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\} \right\}.$$

On pose de plus:

$$\zeta_s^{(t)}(\omega_i) = p(x_s = \omega_i | \mathbf{y}; \boldsymbol{\theta}^{(t)}),$$

$$\begin{aligned} \xi_s^{(t)}(\omega_i, \omega_j) &= p(x_s = \omega_i, x_{s-} = \omega_j | \mathbf{y}; \boldsymbol{\theta}^{(t)}), \\ &= p(x_s = \omega_i | x_{s-} = \omega_j, \mathbf{y}; \boldsymbol{\theta}^{(t)}) p(x_{s-} = \omega_j | \mathbf{y}; \boldsymbol{\theta}^{(t)}). \end{aligned}$$

Expectation Maximization (EM)

$$\boldsymbol{\theta} = \left\{ \left\{ \pi_{\omega_i} \right\}, \left\{ a_{\omega_i, \omega_j} \right\}, \left\{ \mu_{\omega_i} \right\}, \left\{ \sigma_{\omega_i} \right\} \right\}.$$

On pose de plus:

$$\zeta_s^{(t)}(\omega_i) = p(x_s = \omega_i | \mathbf{y}; \boldsymbol{\theta}^{(t)}),$$

$$\begin{aligned} \xi_s^{(t)}(\omega_i, \omega_j) &= p(x_s = \omega_i, x_{s-} = \omega_j | \mathbf{y}; \boldsymbol{\theta}^{(t)}), \\ &= p(x_s = \omega_i | x_{s-} = \omega_j, \mathbf{y}; \boldsymbol{\theta}^{(t)}) p(x_{s-} = \omega_j | \mathbf{y}; \boldsymbol{\theta}^{(t)}). \end{aligned}$$

→ Ces quantités sont **calculables analytiquement** à chaque itération EM à partir des quantités β vues précédemment.

Expectation Maximization (EM)

$$\boldsymbol{\theta} = \left\{ \left\{ \pi_{\omega_i} \right\}, \left\{ a_{\omega_i, \omega_j} \right\}, \left\{ \mu_{\omega_i} \right\}, \left\{ \sigma_{\omega_i} \right\} \right\}.$$

On pose de plus:

$$\zeta_s^{(t)}(\omega_i) = p(x_s = \omega_i | \mathbf{y}; \boldsymbol{\theta}^{(t)}),$$

$$\begin{aligned} \xi_s^{(t)}(\omega_i, \omega_j) &= p(x_s = \omega_i, x_{s-} = \omega_j | \mathbf{y}; \boldsymbol{\theta}^{(t)}), \\ &= p(x_s = \omega_i | x_{s-} = \omega_j, \mathbf{y}; \boldsymbol{\theta}^{(t)}) p(x_{s-} = \omega_j | \mathbf{y}; \boldsymbol{\theta}^{(t)}). \end{aligned}$$

→ Ces quantités sont **calculables analytiquement** à chaque itération EM à partir des quantités β vues précédemment.

Ainsi:

$$\begin{aligned} Q(\boldsymbol{\theta} | \boldsymbol{\theta}^{(t)}) &= \sum_{\omega_i} \zeta_r^{(t)}(\omega_i) \log \pi_{\omega_i} + \sum_{s \in \bar{\mathcal{S}}} \sum_{\omega_i, \omega_j} \xi_s^{(t)}(\omega_i, \omega_j) \log a_{\omega_i, \omega_j} + \\ &\quad \sum_{s \in \mathcal{S}^N} \sum_{\omega_i} \zeta_s^{(t)}(\omega_i) \log f_{\mu_{\omega_i}, \sigma_{\omega_i}}(y_s). \end{aligned}$$

Expectation Maximization (EM)

La maximisation par rapport à chacun des paramètres (avec les contraintes $\sum_{\omega_i} \pi_{\omega_i} = 1$ et $\sum_{\omega_j} a_{\omega_i, \omega_j} = 1$) donne :

$$\pi_{\omega_i}^{(t+1)} = \zeta_r^{(t)}(\omega_i),$$

$$a_{\omega_j, \omega_i}^{(t+1)} = \frac{\sum_{s \in \mathcal{S}} \zeta_s^{(t)}(\omega_i, \omega_j)}{\sum_{s \in \mathcal{S}} \zeta_s^{(t)}(\omega_j)},$$

$$\mu_{\omega_i}^{(t+1)} = \frac{\sum_{s \in \mathcal{S}} \zeta_s^{(t)}(\omega_i) y_s}{\sum_{s \in \mathcal{S}} \zeta_s^{(t)}(\omega_i)},$$

$$\sigma_{\omega_i}^{(t+1)} = \left(\frac{\sum_{s \in \mathcal{S}} \zeta_s^{(t)}(\omega_i) (y_s - \mu_{\omega_i}^{(t+1)})^2}{\sum_{s \in \mathcal{S}} \zeta_s^{(t)}(\omega_i)} \right)^{\frac{1}{2}}.$$

→ Ce sont les **formules de mise à jour** de l'algorithme EM appliqué aux arbres de Markov.

Expectation Maximization (EM)

Algorithme 1 : Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\}..$

Données : θ^0 , un ensemble de paramètres initiaux,
 y , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que *la convergence n'est pas atteinte* **faire**

|

fin

Expectation Maximization (EM)

Algorithme 1 : Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\} \dots$

Données : θ^0 , un ensemble de paramètres initiaux,
 y , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que *la convergence n'est pas atteinte* **faire**

/ Étape E */*

 Calcul de $Q(\theta|\theta^{t-1}) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}|\mathbf{Y}=\mathbf{y}; \theta^{t-1})}[\log p(\mathbf{x}, \mathbf{y}; \theta)]$.

fin

Expectation Maximization (EM)

Algorithme 1 : Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\}..$

Données : θ^0 , un ensemble de paramètres initiaux,
 \mathbf{y} , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que *la convergence n'est pas atteinte* **faire**

 /* Étape E */

 Calcul de $Q(\theta|\theta^{t-1}) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x}|\mathbf{Y}=\mathbf{y}; \theta^{t-1})}[\log p(\mathbf{x}, \mathbf{y}; \theta)]$.

 /* Étape M */

Calcul exact des $\zeta_s^{(t)}(\omega_i)$ et $\xi_s^{(t)}(\omega_i, \omega_j)$ avec les quantités β

fin

Expectation Maximization (EM)

Algorithme 1 : Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\} \dots$

Données : θ^0 , un ensemble de paramètres initiaux,
 y , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que la convergence n'est pas atteinte **faire**

 /* Étape E */

 Calcul de $Q(\theta | \theta^{t-1}) = \mathbb{E}_{\mathbf{x} \sim p(\mathbf{x} | \mathbf{Y} = \mathbf{y}; \theta^{t-1})} [\log p(\mathbf{x}, \mathbf{y}; \theta)]$.

 /* Étape M */

Calcul exact des $\zeta_s^{(t)}(\omega_i)$ et $\xi_s^{(t)}(\omega_i, \omega_j)$ avec les quantités β

 Mise à jour des paramètres $\pi_{\omega_i}^{(t)}, a_{\omega_j, \omega_i}^{(t)}, \mu_{\omega_i}^{(t)}, \sigma_{\omega_i}^{(t)}$

$t \leftarrow t + 1$

fin

Stochastic Expectation Maximization

Algorithme 2 : Stochastic Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\}$.

Données : θ^0 , un ensemble de paramètres initiaux,
 \mathbf{y} , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que *la convergence n'est pas atteinte* **faire**

|

fin

Stochastic Expectation Maximization

Algorithme 2 : Stochastic Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\}$.

Données : θ^0 , un ensemble de paramètres initiaux,
 \mathbf{y} , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que *la convergence n'est pas atteinte* **faire**

 /* Étape E stochastique */

Calcul exact des $p(\mathbf{X}|\mathbf{Y} = \mathbf{y}; \theta^{t-1})$ avec les quantités β .

fin

Stochastic Expectation Maximization

Algorithme 2 : Stochastic Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\}$.

Données : θ^0 , un ensemble de paramètres initiaux,
 \mathbf{y} , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que *la convergence n'est pas atteinte* **faire**

 /* Étape E stochastique */

Calcul exact des $p(\mathbf{X}|\mathbf{Y} = \mathbf{y}; \theta^{t-1})$ avec les quantités β .

 Tirage des échantillons *a posteriori*: $\mathbf{x}^t \sim p(\mathbf{X}|\mathbf{Y} = \mathbf{y}; \theta^{t-1})$.

fin

Stochastic Expectation Maximization

Algorithme 2 : Stochastic Expectation Maximization. Dans le cas des arbres de Markov cachés à bruit gaussien indépendant : $\theta = \{\{\pi_{\omega_i}\}, \{a_{\omega_i, \omega_j}\}, \{\mu_{\omega_i}\}, \{\sigma_{\omega_i}\}\}$.

Données : θ^0 , un ensemble de paramètres initiaux,
 \mathbf{y} , les observations.

Résultat : $\hat{\theta}$, les paramètres estimés.

$t \leftarrow 1$

tant que *la convergence n'est pas atteinte* **faire**

 /* Étape E stochastique */

Calcul exact des $p(\mathbf{X}|\mathbf{Y} = \mathbf{y}; \theta^{t-1})$ avec les quantités β .

 Tirage des échantillons *a posteriori*: $\mathbf{x}^t \sim p(\mathbf{X}|\mathbf{Y} = \mathbf{y}; \theta^{t-1})$.

 /* Étape M */

 Estimateurs Maximum de Vraisemblance pour θ^t avec $(\mathbf{x}^t, \mathbf{y})$.

$t \leftarrow t + 1$

fin

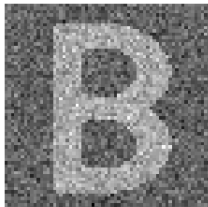
Segmentations non-supervisées

Ex. 1 : arbre simulé et bruit gaussien indépendant "faible"

X



Y
 $\mathcal{N}(1, 1) / \mathcal{N}(4, 1)$



Arbre de Markov χ^{MPM}
err = 0.024



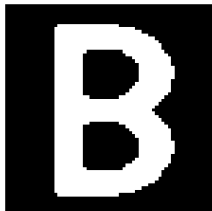
Champ de Markov χ^{MPM}
err = 0.008



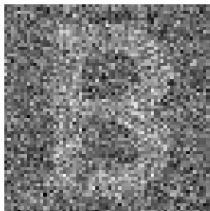
Segmentations non-supervisées

Ex. 2 : arbre simulé et bruit gaussien indépendant "fort"

X



Y
 $\mathcal{N}(1, 1) / \mathcal{N}(2, 1)$



Arbre de Markov X^{MPM}
err = 0.295



Champ de Markov X^{MPM}
err = 0.138



Arbres de Markov couples et triplets

Arbres de Markov couples et triplets

Modèles couples :

Le couple (\mathbf{X}, \mathbf{Y}) forme un arbre de Markov caché :

$$p(\mathbf{X}, \mathbf{Y}) = p(X_r, Y_r) \prod_{s \in \bar{\mathcal{S}}} p(Y_s, X_s | X_{s-}, Y_{s-}).$$

Arbres de Markov couples et triplets

Modèles couples :

Le couple (\mathbf{X}, \mathbf{Y}) forme un arbre de Markov caché :

$$p(\mathbf{X}, \mathbf{Y}) = p(X_r, Y_r) \prod_{s \in \bar{\mathcal{S}}} p(Y_s, X_s | X_{s-}, Y_{s-}).$$

- \mathbf{X} n'est plus nécessairement un arbre de Markov.
- \mathbf{X} sachant $\mathbf{Y} = \mathbf{y}$ reste un arbre de Markov (utile pour l'inférence).

→ **Généralisation** du modèle d'arbre de Markov caché classique

Arbres de Markov couples et triplets

Modèles triplets :

Le triplet $(\mathbf{X}, \mathbf{Y}, \mathbf{V})$ forme un arbre de Markov caché :

$$p(\mathbf{X}, \mathbf{Y}, \mathbf{V}) = p(X_r, Y_r, V_r) \prod_{s \in \bar{\mathcal{S}}} p(Y_s, X_s, V_s | X_{s-}, Y_{s-}, V_{s-}).$$

Arbres de Markov couples et triplets

Modèles triplets :

Le triplet $(\mathbf{X}, \mathbf{Y}, \mathbf{V})$ forme un arbre de Markov caché :

$$p(\mathbf{X}, \mathbf{Y}, \mathbf{V}) = p(X_r, Y_r, V_r) \prod_{s \in \tilde{\mathcal{S}}} p(Y_s, X_s, V_s | X_{s-}, Y_{s-}, V_{s-}).$$

- \mathbf{V} est un vecteur aléatoire auxiliaire formant un nouveau processus caché (\mathbf{X}, \mathbf{V}) lors de l'inférence bayésienne.
- (\mathbf{X}, \mathbf{Y}) n'est plus nécessairement un arbre de Markov.
- (\mathbf{X}, \mathbf{V}) sachant $\mathbf{Y} = \mathbf{y}$ reste un arbre de Markov.
- On retrouve finalement les états cachés originaux par :

$$p(\mathbf{X} | \mathbf{Y}) = \sum_{\mathbf{V}} p(\mathbf{X}, \mathbf{V} | \mathbf{Y}).$$

→ **Généralisation** du modèle d'arbre de Markov caché classique et du modèle d'arbre de Markov couple

Conclusion

Conclusion

Le modèle d'arbre de Markov caché se distingue sur plusieurs points :

- il **généralise les chaînes de Markov** pour le traitement de données multidimensionnelles.

Conclusion

Le modèle d'arbre de Markov caché se distingue sur plusieurs points :

- il **généralise les chaînes de Markov** pour le traitement de données multidimensionnelles.
- il conserve des propriétés d'**inférence exacte**, notamment grâce à l'absence de cycles.

Conclusion

Le modèle d'arbre de Markov caché se distingue sur plusieurs points :

- il **généralise les chaînes de Markov** pour le traitement de données multidimensionnelles.
- il conserve des propriétés d'**inférence exacte**, notamment grâce à l'absence de cycles.
- ce sont des structures de données qui donnent lieu à des **calculs rapides et parallélisables**.

Conclusion

Le modèle d'arbre de Markov caché se distingue sur plusieurs points :

- il **généralise les chaînes de Markov** pour le traitement de données multidimensionnelles.
- il conserve des propriétés d'**inférence exacte**, notamment grâce à l'absence de cycles.
- ce sont des structures de données qui donnent lieu à des **calculs rapides et parallélisables**.
- c'est un modèle encore relativement **peu exploré** dans la littérature scientifique.

Références I

Articles d'introduction directement liés au cours :

- *Discrete Markov Image Modeling and Inference on the Quadtree*, Jean-Marc Laferté et al., IEEE Transactions on Image Processing, 2000.
- *Segmentation non-supervisée d'images par arbres de Markov cachés*, Emmanuel Monfrini et al., GRETSI, 1999.
- *Estimation de mélanges généralisés dans les arbres de Markov cachés, application à la segmentation des images de cartons d'orgue de barbarie*, Emmanuel Monfrini et al., Traitement du signal, 2005.
- *Arbres de Markov couple*, Wojciech Pieczynski, CR Académie des Sciences, 2002.
- *Arbres de Markov couple et segmentation non supervisée d'images : cas de la mono-résolution*, Emmanuel Monfrini et al., GRETSI, 2011.
- *Arbres de Markov triplets pour la segmentation d'images*, Jean-Baptiste Courbot et al., GRETSI 2017.