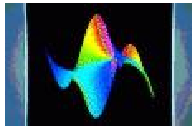
Estimation des arbres de Contextes via les Critères BIC et MDL .

Bezza Hafidi, Véronique Maume-Deschamps

Institut de Mathématiques de Bourgogne

22 Mars, 2007

ALEA 2007



Plan

- Introduction

Plan

Plan

- Introduction
- Algorithme de contexte
- Critères de sélection de Modèle

Plan

- Introduction
- Algorithme de contexte
- Critères de sélection de Modèle
- Résultats

Problématique

Soit A un alphabet fini, $x_1^n = x_1, \ldots, x_n$ une suite finie de mots dans A.

Comment construire un modèle qui pourra produire cette suite ?

Problématique

Soit A un alphabet fini, $x_1^n = x_1, ..., x_n$ une suite finie de mots dans A.

Comment construire un modèle qui pourra produire cette suite ?

Chaine de Markov : chaine d'ordre $k \Rightarrow$ une matrice $|A|^k \times |A|$ et r^{k+1} paramètres à estimer. ($r = |A|$).

Problématique

Soit A un alphabet fini, $x_1^n = x_1, ..., x_n$ une suite finie de mots dans A.

Comment construire un modèle qui pourra produire cette suite ?

Chaine de Markov : chaine d'ordre $k \Rightarrow$ une matrice $|A|^k \times |A|$ et r^{k+1} paramètres à estimer. ($r = |A|$).

VLMC = Chaine de Markov à longueur variable.

Idée : étant donné une chaine de Markov d'ordre k, la connaissance de toute mémoire de longueur k n'est pas nécessaire $\Rightarrow$ arbres de contextes.

Problématique

Soit A un alphabet fini, $x_1^n = x_1, \ldots, x_n$ une suite finie de mots dans A.
Comment construire un modèle qui pourra produire cette suite ?
Chaine de Markov : chaine d'ordre $k \Rightarrow$ une matrice $|A|^k \times |A|$ et r^{k+1} paramètres à estimer. ($r = |A|$).
VLMC = Chaine de Markov à longueur variable.
Idée : étant donné une chaine de Markov d'ordre k, la connaissance de toute mémoire de longueur k n'est pas nécessaire $\Rightarrow$ arbres de contextes.
VLMC introduites par Rissanen en utilisant la thèorie de l'information.
Récemment, étudiées par Bülman and Wyner du point de vue statistique.

Contexte

Soit $(X_n)_{n\in\mathbb{Z}}$ un processus stationnaire, prenant ses valeurs dans A,

Contexte :

Un mot fini x_{-k}^{-1} de longueur minimal tel que pour tout $a \in A$,

$$\begin{aligned}\mathbb{P}(X_0 = a \mid X_{-\infty}^{-1} = x_{-\infty}^{-1}) &= \mathbb{P}(X_0 = a \mid X_{-k}^{-1} = x_{-k}^{-1}) \\ &= \mathbb{P}(X_0 = a \mid X_{-k} = x_{-k}, \ldots, X_{-1} = x_{-1}) \overset{\text{def}}{=} p(a \mid x_{-k}^{-1}). \quad (1)\end{aligned}$$

Contexte

Soit $(X_n)_{n\in\mathbb{Z}}$ un processus stationnaire, prenant ses valeurs dans A,

Contexte :

Un mot fini x_{-k}^{-1} de longueur minimal tel que pour tout $a \in A$,

$$\begin{aligned}\mathbb{P}(X_0 = a \mid X_{-\infty}^{-1} = x_{-\infty}^{-1}) &= \mathbb{P}(X_0 = a \mid X_{-k}^{-1} = x_{-k}^{-1}) \\ &= \mathbb{P}(X_0 = a \mid X_{-k} = x_{-k}, \ldots, X_{-1} = x_{-1}) \overset{\text{def}}{=} p(a \mid x_{-k}^{-1}). \quad (1)\end{aligned}$$

S'il existe, pour tout vecteur du passé $x_{-\infty}^{-1}$, un indice $k = k(x_{-\infty}^{-1})$ tel que (1) est vérifié, alors le processus $(X_n)_{n\in\mathbb{Z}}$ est appelé VLMC.

Contexte

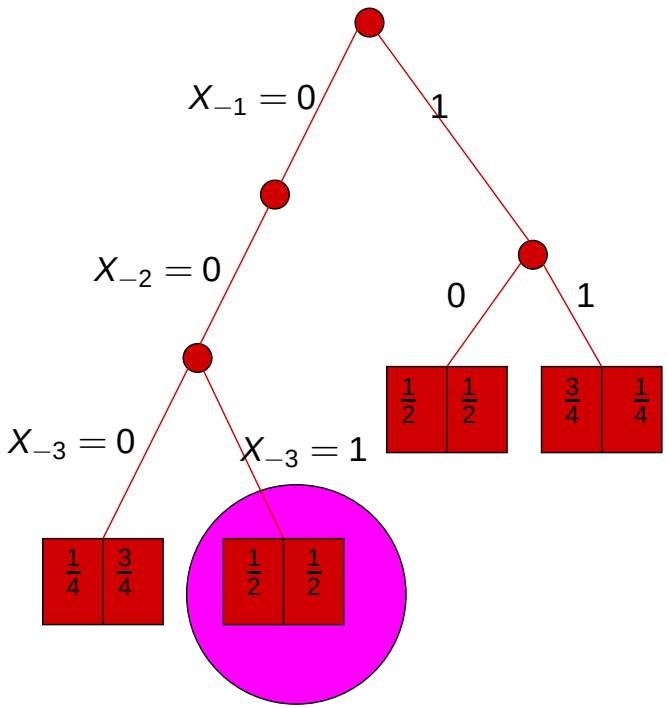
Soit $(X_n)_{n\in\mathbb{Z}}$ un processus stationnaire, prenant ses valeurs dans A,
Contexte :

Un mot fini x_{-k}^{-1} de longueur minimal tel que pour tout $a \in A$,

$$\begin{aligned}\mathbb{P}(X_0 = a \mid X_{-\infty}^{-1} = x_{-\infty}^{-1}) &= \mathbb{P}(X_0 = a \mid X_{-k}^{-1} = x_{-k}^{-1}) \\ &= \mathbb{P}(X_0 = a \mid X_{-k} = x_{-k}, \ldots, X_{-1} = x_{-1}) \overset{\text{def}}{=} p(a \mid x_{-k}^{-1}). \quad (1)\end{aligned}$$

S'il existe, pour tout vecteur du passé $x_{-\infty}^{-1}$, un indice $k = k(x_{-\infty}^{-1})$ tel que (1) est vérifié, alors le processus $(X_n)_{n\in\mathbb{Z}}$ est appelé VLMC.
τ est l'ensemble de tous les contextes de la VLMC, propriété du suffixe
$\Rightarrow$ peut être représenté par un arbre.

Arbre de contexte, VLMC

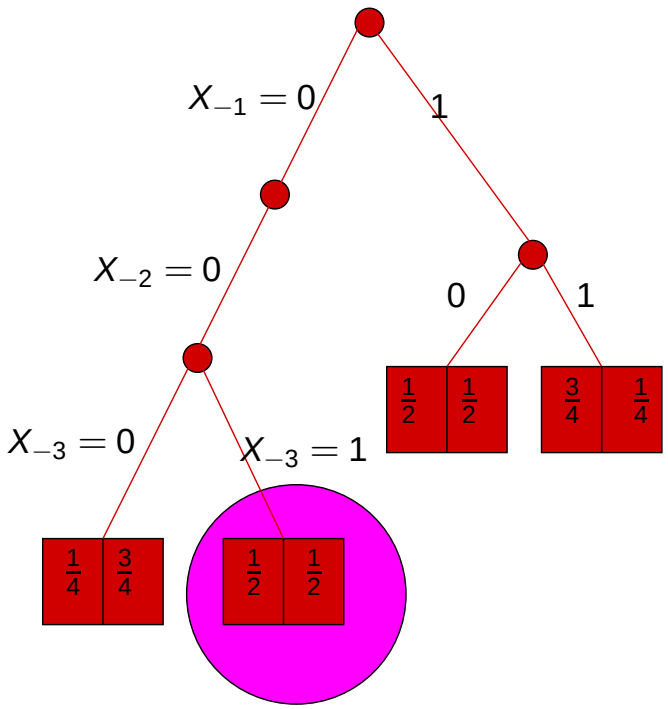


Soit $A = \{0, 1\}$
Les mots : 000, 100, 01, 11 sont des contextes.

$$\mathbb{P}(X_0 = 1 | X_{-3} = 1,\ X_{-2} = 0,\ X_{-1} = 0)$$
$$= \quad p(1|100) = \frac{1}{2},$$

$$p(0|100) = \tfrac{1}{2},\ p(1|011) = p(1|11).$$

Arbre de contexte, VLMC



Soit $A = \{0, 1\}$
Les mots : 000, 100, 01, 11 sont des contextes.

$$\mathbb{P}(X_0 = 1 | X_{-3} = 1,\ X_{-2} = 0,\ X_{-1} = 0)$$
$$= \quad p(1|100) = \frac{1}{2},$$

$p(0|100) = \frac{1}{2}$, $p(1|011) = p(1|11)$.

Estimation de (τ, p_τ) ?

Probabilités empirique

Soit $(X_1, \ldots, X_n)$ un échantillon fini et w un caractère de longueur inférieur à n.

$N_n(w)$ est le nombre d'occurences de w dans la séquence $(X_1, \ldots, X_n)$:

$$N_n(w) = \sum_{m=1}^{n-|w|} \mathbf{1}\{X_m^{m+|w|-1} = w\}.$$

Probabilités empirique

Soit $(X_1, \ldots, X_n)$ un échantillon fini et w un caractère de longueur inférieur à n.

$N_n(w)$ est le nombre d'occurences de w dans la séquence $(X_1, \ldots, X_n)$:

$$N_n(w) = \sum_{m=1}^{n-|w|} \mathbf{1}\{X_m^{m+|w|-1} = w\}.$$

La probabilité de transition empirique $\hat{p}_n(a|w)$ est définie par :

$$\hat{p}_n(a|w) = \frac{N_n(wa)}{N_n(w)}.$$

Algorithme de contexte

Etant donnée deux mots u et w,

$$\Delta_n(u,w) = \sum_{a \in A} \hat{p}_n(a|wu) \log \left(\frac{\hat{p}_n(a|wu)}{\hat{p}_n(a|w)} \right) N_n(wu)$$

est la mesure de l'information fournie par u relativement à w.

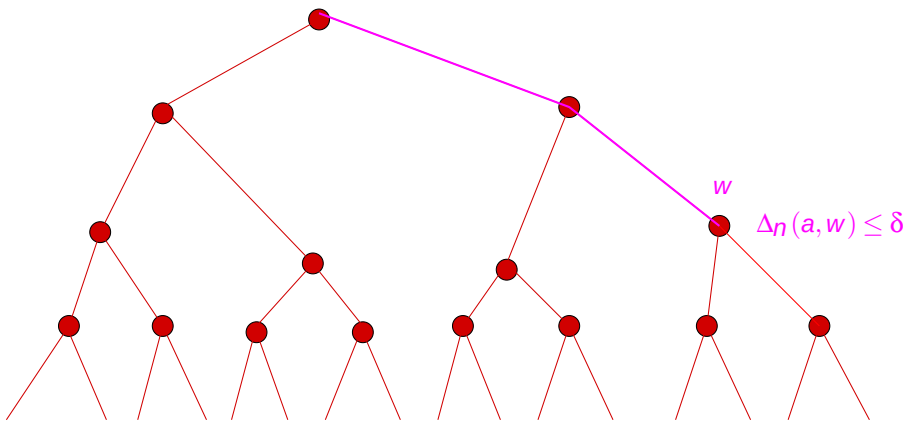
Remarque : si w est contexte alors, pour tout u,

$$p_\tau(a|w) = p_\tau(a|uw).$$

Algorithme de contexte II

Algorithme : considère tous les caractères de longueur ℓ, comme un arbre, élaguer l'arbre comme le suivant : calculer $\Delta_n(a, w)$, pour tout noeud w et une lettre a. Si $\Delta_n(a, w) \leq \delta$, couper tous les sous-arbres.

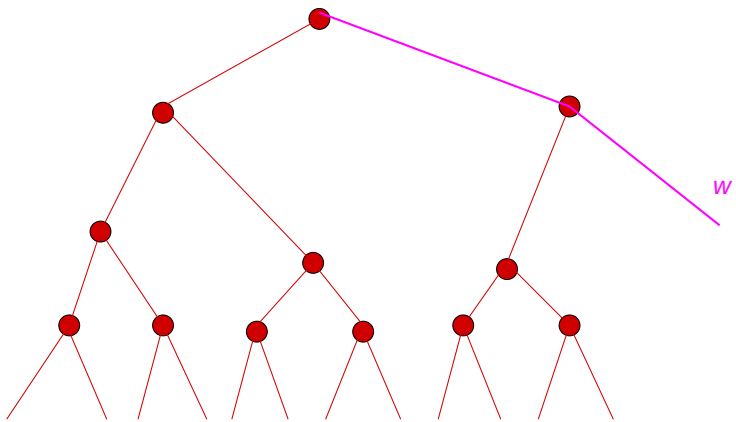
$$\Delta_n(a, w) \leq \delta.$$



Algorithme de contexte II

Algorithme : considère tous les caractères de longueur ℓ, comme un arbre, élaguer l'arbre comme le suivant : calculer $\Delta_n(a, w)$, pour tout noeud w et une lettre a. Si $\Delta_n(a, w) \leq \delta$, couper tous les sous-arbres.

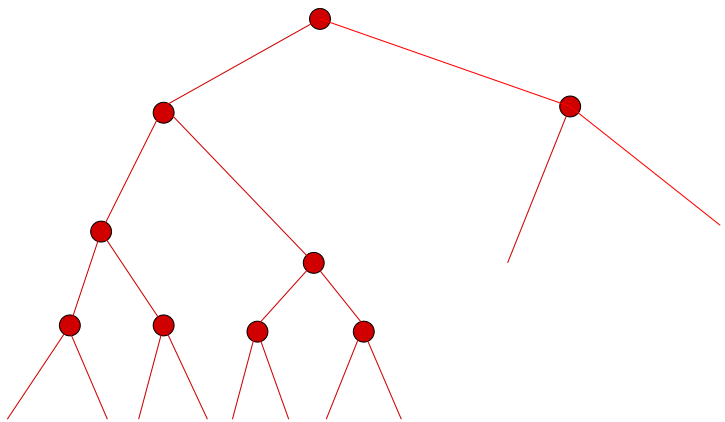
$$\Delta_n(a, w) \leq \delta.$$



Algorithme de contexte II

Algorithme : considère tous les caractères de longueur ℓ, comme un arbre, élaguer l'arbre comme le suivant : calculer $\Delta_n(a, w)$, pour tout noeud w et une lettre a. Si $\Delta_n(a, w) \leq \delta$, couper tous les sous-arbres.

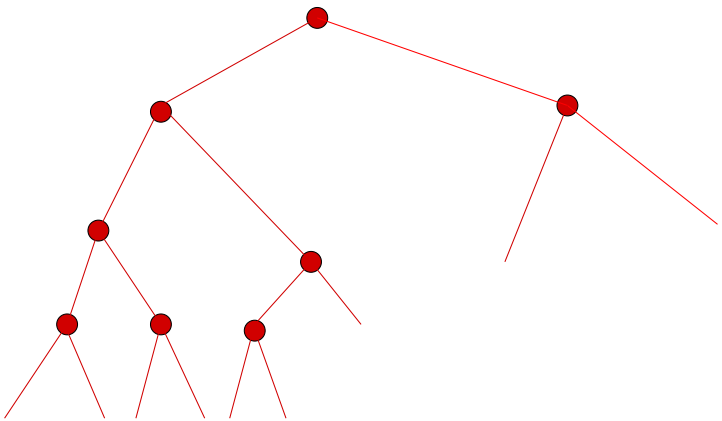
$$\Delta_n(a, w) \leq \delta.$$



Algorithme de contexte II

Algorithme : considère tous les caractères de longueur ℓ, comme un arbre, élaguer l'arbre comme le suivant : calculer $\Delta_n(a, w)$, pour tout noeud w et une lettre a. Si $\Delta_n(a, w) \leq \delta$, couper tous les sous-arbres.

$$\Delta_n(a, w) \leq \delta.$$

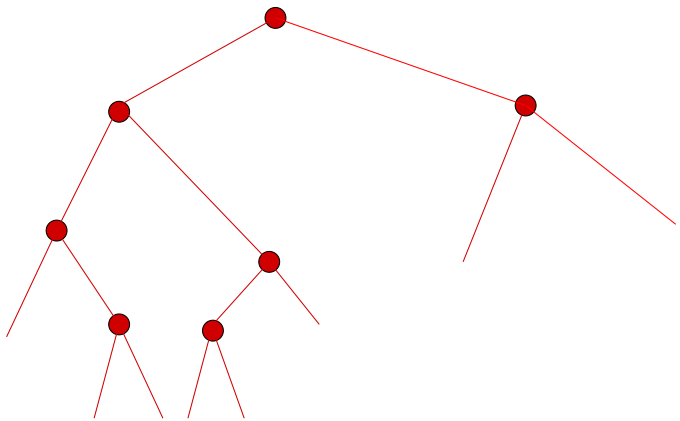


Algorithme de contexte II

Algorithme : considère tous les caractères de longueur ℓ, comme un arbre, élaguer l'arbre comme le suivant : calculer $\Delta_n(a,w)$, pour tout noeud w et une lettre a. Si $\Delta_n(a,w) \leq \delta$, couper tous les sous-arbres.

$$\Delta_n(a,w) \leq \delta.$$

Libeler les feuilles par les probabilités empiriques $\hat{p}_n(a|w)$.



Arbre empirique

Soit $\delta > 0$, $\ell \in \mathbb{N}$,
Arbre empirique : $\hat{\tau}_n(\ell)$ = l'ensemble maximum de tous les caractères finis x_{-k}^{-1}, $1 \leq k \leq \ell$ tq pour tout $j \leq k$

$$\Delta_n(x_{-j}, x_{-j+1}^{-1}) > \delta.$$

Comment choisir δ, ℓ, tq $\hat{\tau}_n$ approxime τ ? Dans quel sens $(\hat{\tau}_n(\ell), \hat{p}_n)$ approxime (τ, p_τ) ?

Critère BIC

Sélection de modèle : trouver l'arbre qui minimise certains critères d'information.

$$\tilde{P}_{ML,w}(x_1^n) = \prod_{a \in A} \left(\frac{N_n(w,a)}{N_n(w)} \right)^{N_n(w,a)}, \tilde{P}_w(x_1^n) = n^{-\frac{|A|-1}{2}} \tilde{P}_{ML,w}(x_1^n),$$

Critère BIC

Sélection de modèle : trouver l'arbre qui minimise certains critères d'information.

$$\tilde{P}_{ML,w}(x_1^n) = \prod_{a \in A} \left(\frac{N_n(w,a)}{N_n(w)} \right)^{N_n(w,a)}, \tilde{P}_w(x_1^n) = n^{-\frac{|A|-1}{2}} \tilde{P}_{ML,w}(x_1^n),$$

$$\mathrm{BIC}_\tau(x_1^n) = - \sum_{w \in \tau, N_n(w) \geq 1} \log \tilde{P}_{ML,w}(x_1^n) + \frac{(|A|-1)|\tau|}{2} \log n.$$

Critère BIC

Sélection de modèle : trouver l'arbre qui minimise certains critères d'information.

$$\tilde{P}_{ML,w}(x_1^n) = \prod_{a \in A} \left(\frac{N_n(w,a)}{N_n(w)} \right)^{N_n(w,a)}, \tilde{P}_w(x_1^n) = n^{-\frac{|A|-1}{2}} \tilde{P}_{ML,w}(x_1^n),$$

$$\mathrm{BIC}_\tau(x_1^n) = -\sum_{w \in \tau, N_n(w) \geq 1} \log \tilde{P}_{ML,w}(x_1^n) + \frac{(|A|-1)|\tau|}{2} \log n.$$

$$\mathrm{BIC}_{Emp_\tau}(x_1^n) = \sum_{w \in \tau, N_n(w) \geq 1} \left\{ -\log \tilde{P}_{ML,w}(x_1^n) + \frac{(|A|-1)}{2} \log N_n(w) \right\}.$$

Critère MDL (KT)

$$\tilde{P}_{KT,w}(x_1^n) = \frac{\prod_{a,N_n(w,a)\geq 1}\left[(N_n(w,a)-\frac{1}{2})(N_n(w,a)-\frac{3}{2})...(\frac{1}{2})\right]}{(N_n(w)-1+\frac{|A|}{2})(N_n(w)-2+\frac{|A|}{2})...(\frac{|A|}{2})}$$

$$\mathrm{KT}_\tau(x_1^n) = -\sum_{w\in\tau,N_n(w)\geq 1} \log \tilde{P}_{KT,w}(x_1^n) + D(n)\log|A|$$

Critère MDL (KT)

$$\tilde{P}_{KT,w}(x_1^n) = \frac{\prod_{a,N_n(w,a)\geq 1}\left[(N_n(w,a)-\frac{1}{2})(N_n(w,a)-\frac{3}{2})...(\frac{1}{2})\right]}{(N_n(w)-1+\frac{|A|}{2})(N_n(w)-2+\frac{|A|}{2})...(\frac{|A|}{2})}$$

$$\mathrm{KT}_\tau(x_1^n) = -\sum_{w\in\tau,N_n(w)\geq 1} \log \tilde{P}_{KT,w}(x_1^n) + D(n)\log|A|$$

$$\hat{\tau}_n = \arg\min_\tau \{BIC_\tau, KT_\tau\}$$

Algorithme BIC, MDL (révisé)

Soit

$$\Delta_w(x_1^n) = \frac{\prod_{a,N_n(w,a)\geq 1} V_{aw}(x_1^n)}{\tilde{P}_w(x_1^n)}.$$

avec $V_{aw} = \prod_{u\in\tau_{aw}} \tilde{P}_u(x_1^n)$

Algorithme BIC, MDL (révisé)

Soit

$$\Delta_w(x_1^n) = \frac{\prod_{a,N_n(w,a)\geq 1} V_{aw}(x_1^n)}{\tilde{P}_w(x_1^n)}.$$

avec $V_{aw} = \prod_{u\in\tau_{aw}} \tilde{P}_u(x_1^n)$

On construit l'arbre complet de profondeur $h(\tau) = c \ln n$, et on calcule pour tout w noeud de l'arbre, $\Delta_w(x_1^n)$.
Si $\Delta_w(x_1^n) \leq \delta$ on coupe le sous-arbre issue de w. l'arbre obtenu minimise le critère BIC ou KT

Algorithme BIC, MDL (révisé)

Soit

$$\Delta_w(x_1^n) = \frac{\prod_{a,N_n(w,a)\geq 1} V_{aw}(x_1^n)}{\tilde{P}_w(x_1^n)}.$$

avec $V_{aw} = \prod_{u\in\tau_{aw}} \tilde{P}_u(x_1^n)$

On construit l'arbre complet de profondeur $h(\tau) = c \ln n$, et on calcule pour tout w noeud de l'arbre, $\Delta_w(x_1^n)$.
Si $\Delta_w(x_1^n) \leq \delta$ on coupe le sous-arbre issue de w. l'arbre obtenu minimise le critère BIC ou KT
Nous *obtenons* les inégalités exponentielles = montre la convergence p.s de cet algorithme.

Paramètres initiaux

Soit

$$T = \max_{\substack{a \in A \\ w \in \tau}} \tau(aw).$$

$$h(\tau) = \max\{|w|; w \in \tau\}.$$

$$\rho = \min_{\substack{a \in A \\ w \in \tau}} p(a|w),$$

$$p_{\min} = \inf_{w \in \tau} p(w)$$

et β est la borne des coefficient de la chaine.

Tous ces paramètres sont supposés > 0.

Convergence p.s

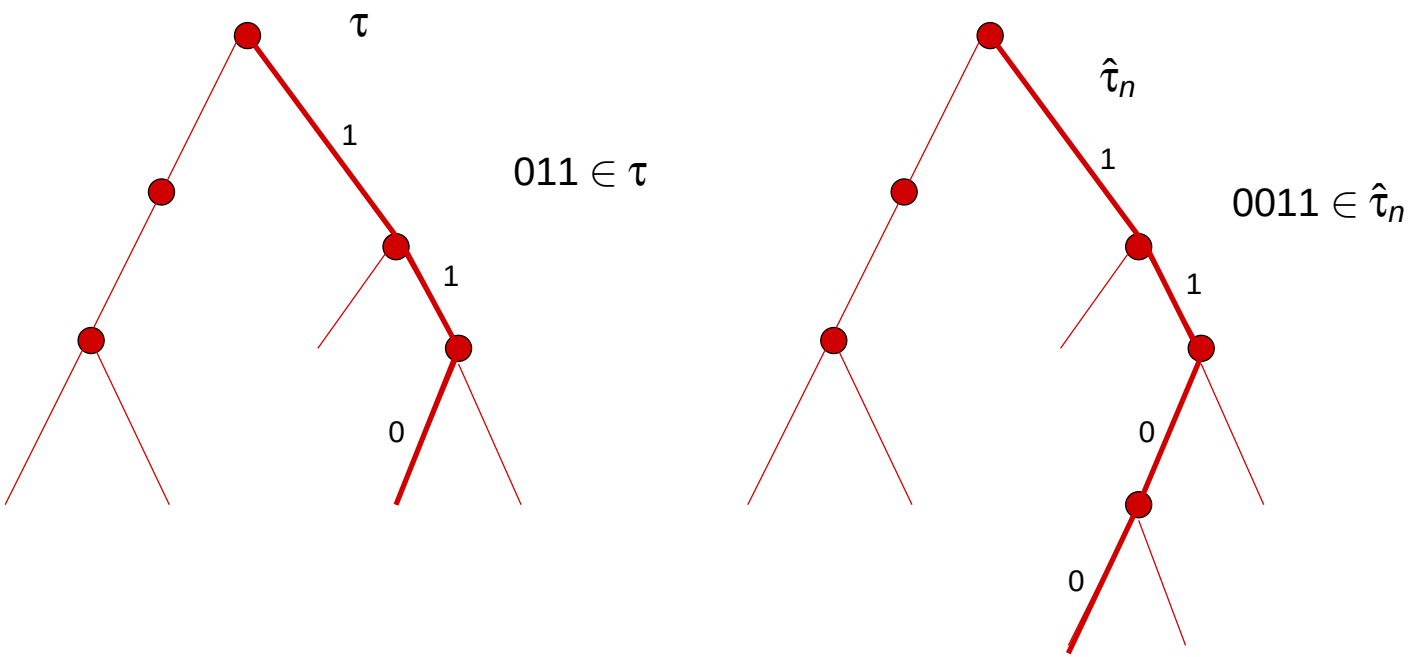
$(X_1, \ldots, X_n)$ une suite de VLMC, de probabilités (τ, p_τ) avec τ est de hauteur $h(\tau) < \infty$. Alors,

Theorem

$$P(\hat{\tau}_{Bic,KT} = \tau) \quad \geq \quad 1 - (K_1 + K_2 + K_3)$$

Sur-estimations

$\hat{\tau} \geq \tau$ s'il existe un contexte $w \in \tau$ tq : $uw \in \hat{\tau}$.



Inégalité exponentielle, BIC

$(X_1, \ldots, X_n)$ une suite de VLMC, de probabilités (τ, p_τ) avec τ est de hauteur $h(\tau) < \infty$. Alors,

Proposition

$$P(\hat{\tau}_{Bic} > \tau) \leq$$
$$|A|^{2h(\tau)} T.e^{1/e} \left(2\exp\left\{ -\frac{(t - \frac{|A|+1}{np_{min}})^2 np_{min}^2 \beta}{8e} \right\} \right.$$
$$\left. + \exp\left\{ -\beta \frac{np_{min}^2}{8e} \right\} \right)$$

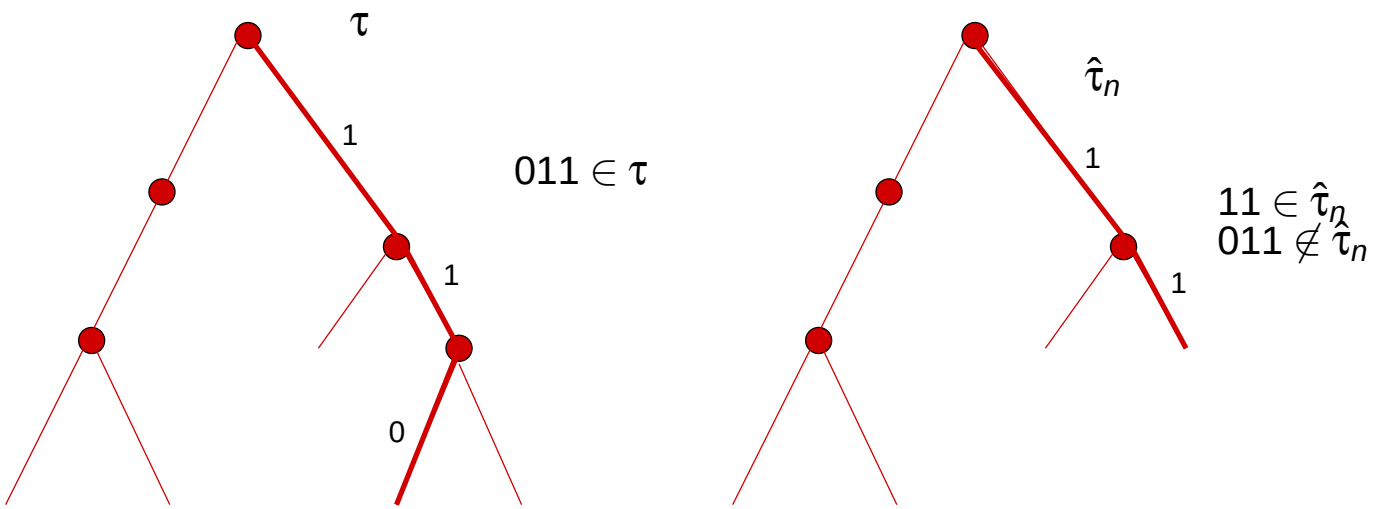
Inégalité exponentielle, BIC Empirique et KT

Proposition

$$P(\hat{\tau}_{BIC_{Emp},KT} > \tau) \leq$$
$$2|A|^{2h(\tau)} T.e^{1/e} \left[\exp\left\{ -\frac{(t - \frac{|A|+1}{np_{min}})^2 np_{min}^2 \beta}{18e} \right\} \right.$$
$$\left. + \frac{1}{2} \exp\left\{ -\beta \frac{np_{min}^2}{8e} \right\} + \frac{1}{2} \exp\left\{ -\beta \frac{np_{min}^2}{8e} \right\} \right]$$

Sous-estimations

$\hat{\tau} \leq \tau$ s'il existe $s \in \hat{\tau}$ tq s est un suffixe d'un context $w = us \in \tau$.



Inégalité exponentielle, BIC

Proposition

$$P(\hat{\tau}_{BIC} < \tau) \leq$$
$$2|A|^{2h(\tau)} T.e^{1/e} \left[\exp\left\{ -\frac{(t - \frac{|A|+1}{np_{min}})^2 np_{min}^2 \beta}{8e} \right\} \right.$$
$$+ \exp\left\{ -\frac{(t - \frac{|A|+1}{np_{min}})^2 np_{min}^2 \beta}{8e} \right\}$$
$$\left. + \frac{1}{2} \exp\left\{ -\beta \frac{np_{min}^2}{8e} \right\} \right]$$

Inégalité exponentielle, BIC Empirique et KT

Proposition

$$P(\hat{\tau}_{BIC_{Emp},KT} < \tau) \leq$$
$$2|A|^{2h(\tau)} T.e^{1/e} \left[\exp\left\{ -\frac{(t - \frac{|A|+1}{np_{min}})^2 np_{min}^2 \beta}{8e} \right\} \right.$$
$$+ \exp\left\{ -\frac{(t - \frac{|A|+1}{np_{min}})^2 np_{min}^2 \beta}{8e} \right\}$$
$$\left. + \frac{1}{2} \exp\left\{ -\beta \frac{np_{min}^2}{8e} \right\} + \frac{1}{2} \exp\left\{ -\beta \frac{np_{min}^2}{8e} \right\} \right]$$

Perspective

- Estimer le seuil δ en utilisant le critère validation croisée.
- Application des arbres de contextes pour prédire les séquences bilogique, tq ADN..
- Comparer différents algorithmes pour estimer les arbres de contextes.

Références

- P. Bühlmann and A. Wyner (1999), *Variable length Markov chains.* Ann. Statist. **27**, no. 2, 480–513.
- F. Ferrari and A. Wyner (2003), *Estimation of general stationary processes by variable length Markov chains*, Scand. J.e Statist. **30**, no. 3, 459–480.
- J. Rissanen (1983) *A universal data compression system.* IEEE Trans. Inform. Theory **29**, no. 5, 656–664.
- I. Csiszár and Z.Talata (2005), *Context tree estimation for not necessarily finite memory processes via BIC and MDL*, IEEE Transactions on Information Theory, Vol.52, No.3, pp. 1007–1016, Mar 2006.
- V.M Deschamps, A.Galves and B.Schmitt (2006). *Exponential inequalities for VLMC empirical trees.* preprint