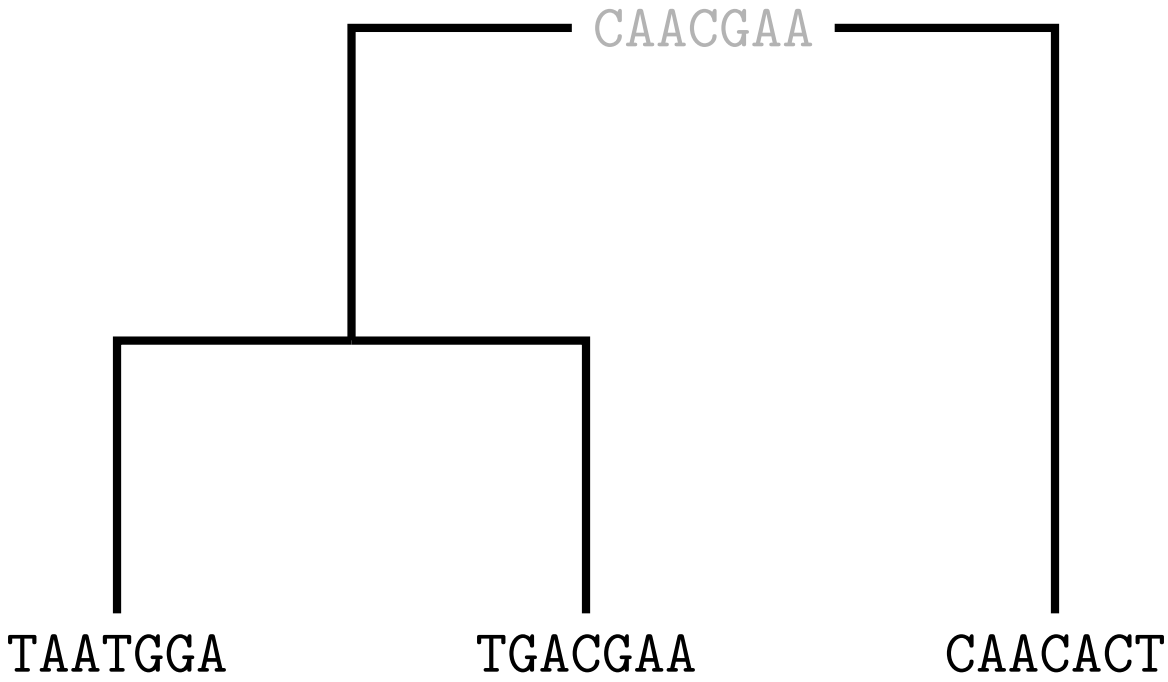
Méthodes particulaires et vraisemblances pour l'inférence de modèles d'évolution avec dépendance au contexte

Alexis Huet

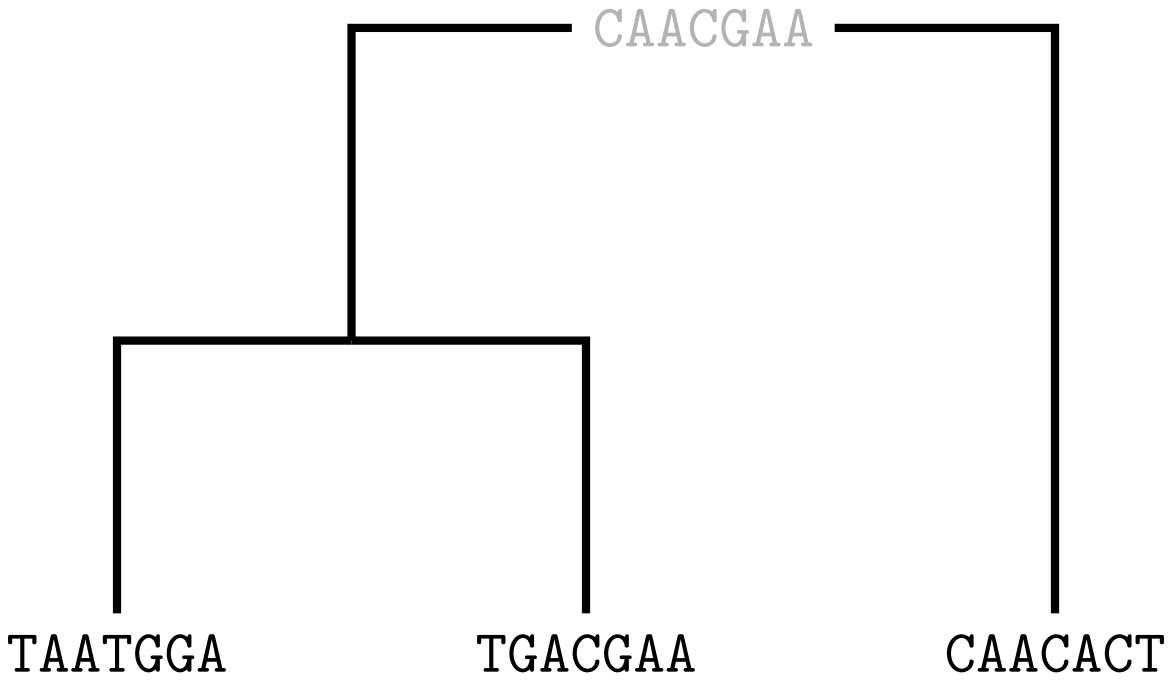
27 juin 2014

Plan

Plan



$\rightarrow$ Problématique : étant donné un modèle d'évolution et une loi pour la séquence ancestrale, calculer la vraisemblance d'un alignement de séquences actuelles.



$\rightarrow$ Problématique : étant donné un modèle d'évolution et une loi pour la séquence ancestrale, calculer la vraisemblance d'un alignement de séquences actuelles.

Modèles à sites indépendants

- chaque site évolue de façon indépendante selon la même loi,
- chaîne de Markov en temps continu sur $\{A, C, G, T\}$,

Exemple : générateur pour les modèles RN95 (Rzhetsky / Nei)

$$Q = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{c} \begin{array}{cccc} A & C & G & T \end{array} \\ \begin{pmatrix} . & v_C & w_G & v_T \\ v_A & . & v_G & w_T \\ w_A & v_C & . & v_T \\ v_A & w_C & v_G & . \end{pmatrix} \end{array}$$

Deux types de bases : A, G=purines=R et C, T=pyrimidines=Y.
Transversion : substitution $R \to Y$ ou $Y \to R$.
Transition : substitution $R \to R$ ou $Y \to Y$.

Modèles à sites indépendants

- chaque site évolue de façon indépendante selon la même loi,
- chaîne de Markov en temps continu sur $\{A, C, G, T\}$,

Exemple : générateur pour les modèles RN95 (Rzhetsky / Nei)

$$Q = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{c} \begin{array}{cccc} A & C & G & T \end{array} \\ \begin{pmatrix} . & v_C & w_G & v_T \\ v_A & . & v_G & w_T \\ w_A & v_C & . & v_T \\ v_A & w_C & v_G & . \end{pmatrix} \end{array}$$

Deux types de bases : A, G=purines=R et C, T=pyrimidines=Y.
Transversion : substitution $R \to Y$ ou $Y \to R$.
Transition : substitution $R \to R$ ou $Y \to Y$.

Modèles à sites indépendants

- chaque site évolue de façon indépendante selon la même loi,
- chaîne de Markov en temps continu sur $\{A, C, G, T\}$,

Exemple : générateur pour les modèles RN95 (Rzhetsky / Nei)

$$Q = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{c} \begin{array}{cccc} A & C & G & T \end{array} \\ \begin{pmatrix} . & v_C & w_G & v_T \\ v_A & . & v_G & w_T \\ w_A & v_C & . & v_T \\ v_A & w_C & v_G & . \end{pmatrix} \end{array}$$

Deux types de bases : $A, G =$ purines $= R$ et $C, T =$ pyrimidines $= Y$.

Transversion : substitution $R \to Y$ ou $Y \to R$.

Transition : substitution $R \to R$ ou $Y \to Y$.

Modèles à sites indépendants

- chaque site évolue de façon indépendante selon la même loi,
- chaîne de Markov en temps continu sur $\{A, C, G, T\}$,

Exemple : générateur pour les modèles RN95 (Rzhetsky / Nei)

$$Q = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{c} \begin{array}{cccc} A & C & G & T \end{array} \\ \begin{pmatrix} . & v_C & w_G & v_T \\ v_A & . & v_G & w_T \\ w_A & v_C & . & v_T \\ v_A & w_C & v_G & . \end{pmatrix} \end{array}$$

Deux types de bases : $A, G =$ purines $= R$ et $C, T =$ pyrimidines $= Y$.
Transversion : substitution $R \to Y$ ou $Y \to R$.
Transition : substitution $R \to R$ ou $Y \to Y$.

Modèles à sites indépendants

- chaque site évolue de façon indépendante selon la même loi,
- chaîne de Markov en temps continu sur $\{A, C, G, T\}$,

Exemple : générateur pour les modèles RN95 (Rzhetsky / Nei)

$$Q = \begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{c} \begin{array}{cccc} A & C & G & T \end{array} \\ \begin{pmatrix} . & v_C & w_G & v_T \\ v_A & . & v_G & w_T \\ w_A & v_C & . & v_T \\ v_A & w_C & v_G & . \end{pmatrix} \end{array}$$

Deux types de bases : A, G=purines=R et C, T=pyrimidines=Y.
Transversion : substitution $R \to Y$ ou $Y \to R$.
Transition : substitution $R \to R$ ou $Y \to Y$.

Taux de saut

$$P\left(\begin{array}{ll} \ldots \text{ACGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{ACATA} \ldots & \text{temps t+dt} \end{array}\right) = w_A dt + o(dt),$$

$$P\left(\begin{array}{ll} \ldots \text{AGGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{AGATA} \ldots & \text{temps t+dt} \end{array}\right) = w_A dt + o(dt).$$

Taux de saut

$$P\left(\begin{array}{ll} \ldots \text{ACGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{ACATA} \ldots & \text{temps t+dt} \end{array}\right) = w_A dt + o(dt),$$

$$P\left(\begin{array}{ll} \ldots \text{AGGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{AGATA} \ldots & \text{temps t+dt} \end{array}\right) = w_A dt + o(dt).$$

Hypermutabilité des dinucléotides CpG

Biochimiquement (par exemple chez les mammifères) :

- Taux $C \to T$ accru si le nucléotide à droite est G,
- Taux $G \to A$ accru si le nucléotide à gauche est C.

$\to$ Nécessité de définir des taux de substitution prenant en compte le contexte local.

Hypermutabilité des dinucléotides CpG

Biochimiquement (par exemple chez les mammifères) :

- Taux $C \to T$ accru si le nucléotide à droite est G,
- Taux $G \to A$ accru si le nucléotide à gauche est C.

$\to$ Nécessité de définir des taux de substitution prenant en compte le contexte local.

Taux de saut avec renforcement

$$P\left(\begin{array}{ll} \ldots \text{ACGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{ACATA} \ldots & \text{temps t+dt} \end{array}\right) = (w_A + r_{CG \to CA})dt + o(dt),$$

$$P\left(\begin{array}{ll} \ldots \text{AGGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{AGATA} \ldots & \text{temps t+dt} \end{array}\right) = w_A dt + o(dt).$$

Taux de saut avec renforcement

$$P\begin{pmatrix} \ldots \text{ACGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{ACATA} \ldots & \text{temps t+dt} \end{pmatrix} = (w_A + r_{CG \to CA})dt + o(dt),$$

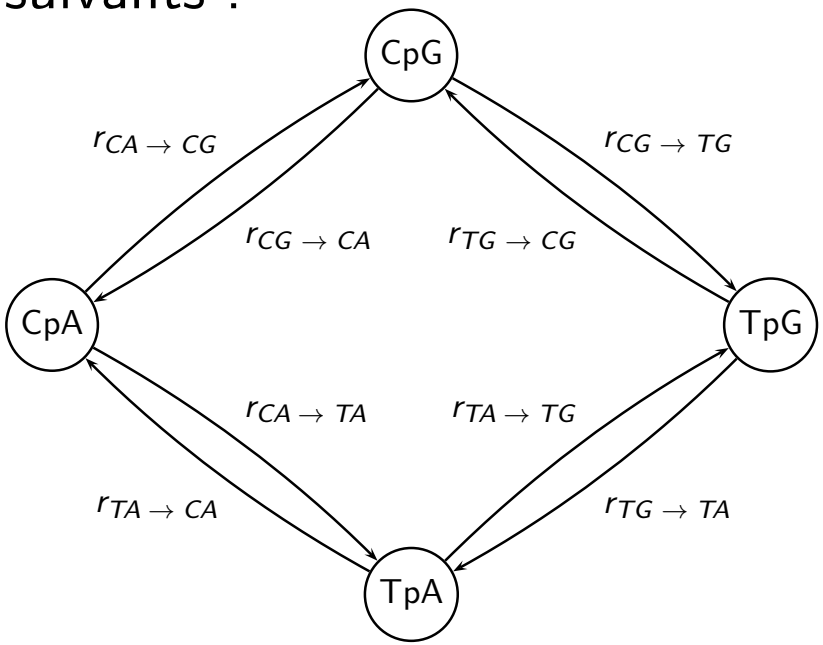
$$P\begin{pmatrix} \ldots \text{AGGTA} \ldots & \text{temps t} \\ \downarrow & \\ \ldots \text{AGATA} \ldots & \text{temps t+dt} \end{pmatrix} = w_A dt + o(dt).$$

Classe de modèles RN95+YpR

- modèle RN95 de matrice de sauts

$$\begin{array}{c} \\ A \\ C \\ G \\ T \end{array} \begin{array}{c} \begin{array}{cccc} A & C & G & T \end{array} \\ \begin{pmatrix} . & v_C & w_G & v_T \\ v_A & . & v_G & w_T \\ w_A & v_C & . & v_T \\ v_A & w_C & v_G & . \end{pmatrix} \end{array},$$

- renforcements suivants :



Apports de la thèse

- Mise en évidence de structures de dépendances markoviennes.
- Introduction de méthodes particulaires pour calculer les vraisemblances en exploitant ces structures :
  - création de code (C++),
  - applications.
- Comportement asymptotique d'estimateurs :
  - consistance et normalité asymptotique du maximum de vraisemblance,
  - estimateur semi-empirique de la variance pour la méthode d'estimation de [BG12].
- Introduction d'une vraisemblance composite « approximation markovienne ».
- Étude théorique de cas limites (mise en évidence des phénomènes de dépendance).

Apports de la thèse

- Mise en évidence de structures de dépendances markoviennes.
- Introduction de méthodes particulaires pour calculer les vraisemblances en exploitant ces structures :
  - création de code (C++),
  - applications.
- Comportement asymptotique d'estimateurs :
  - consistance et normalité asymptotique du maximum de vraisemblance,
  - estimateur semi-empirique de la variance pour la méthode d'estimation de [BG12].
- Introduction d'une vraisemblance composite « approximation markovienne ».
- Étude théorique de cas limites (mise en évidence des phénomènes de dépendance).

Apports de la thèse

- Mise en évidence de structures de dépendances markoviennes.
- Introduction de méthodes particulaires pour calculer les vraisemblances en exploitant ces structures :
  - création de code (C++),
  - applications.
- Comportement asymptotique d'estimateurs :
  - consistance et normalité asymptotique du maximum de vraisemblance,
  - estimateur semi-empirique de la variance pour la méthode d'estimation de [BG12].
- Introduction d'une vraisemblance composite « approximation markovienne ».
- Étude théorique de cas limites (mise en évidence des phénomènes de dépendance).

Apports de la thèse

- Mise en évidence de structures de dépendances markoviennes.
- Introduction de méthodes particulaires pour calculer les vraisemblances en exploitant ces structures :
  - création de code (C++),
  - applications.
- Comportement asymptotique d'estimateurs :
  - consistance et normalité asymptotique du maximum de vraisemblance,
  - estimateur semi-empirique de la variance pour la méthode d'estimation de [BG12].
- Introduction d'une vraisemblance composite « approximation markovienne ».
- Étude théorique de cas limites (mise en évidence des phénomènes de dépendance).

Apports de la thèse

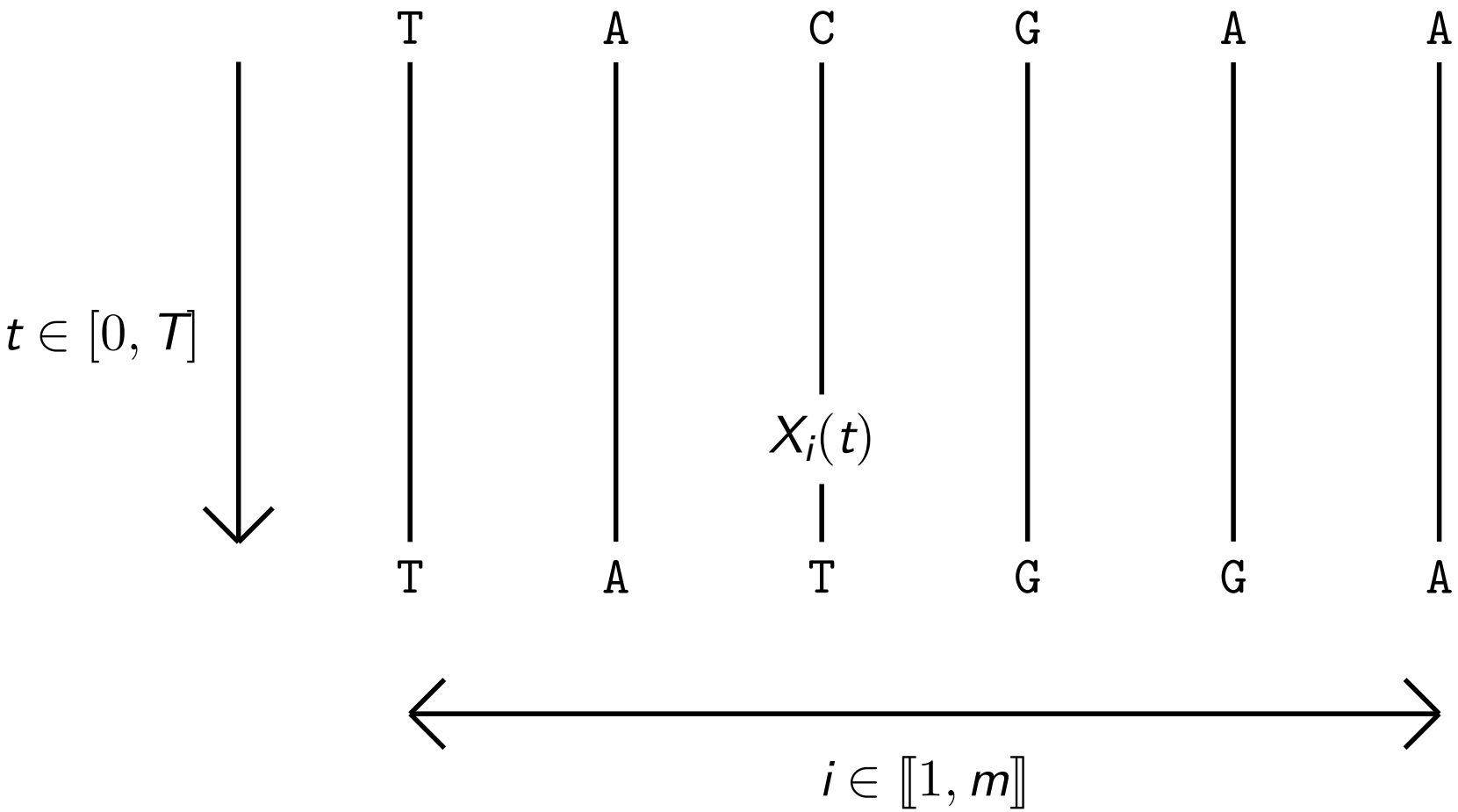
- Mise en évidence de structures de dépendances markoviennes.
- Introduction de méthodes particulaires pour calculer les vraisemblances en exploitant ces structures :
  - création de code (C++),
  - applications.
- Comportement asymptotique d'estimateurs :
  - consistance et normalité asymptotique du maximum de vraisemblance,
  - estimateur semi-empirique de la variance pour la méthode d'estimation de [BG12].
- Introduction d'une vraisemblance composite « approximation markovienne ».
- Étude théorique de cas limites (mise en évidence des phénomènes de dépendance).

Apports de la thèse

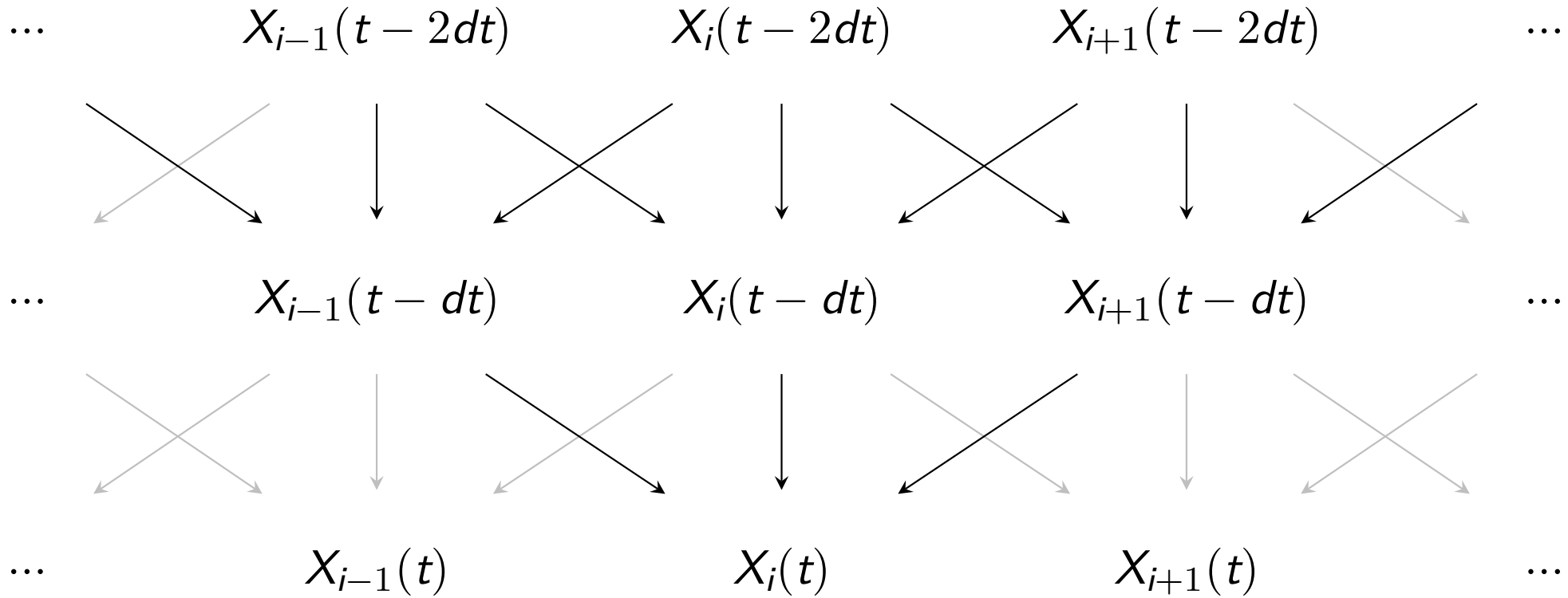
- Mise en évidence de structures de dépendances markoviennes.
- Introduction de méthodes particulaires pour calculer les vraisemblances en exploitant ces structures :
  - création de code (C++),
  - applications.
- Comportement asymptotique d'estimateurs :
  - consistance et normalité asymptotique du maximum de vraisemblance,
  - estimateur semi-empirique de la variance pour la méthode d'estimation de [BG12].
- Introduction d'une vraisemblance composite « approximation markovienne ».
- Étude théorique de cas limites (mise en évidence des phénomènes de dépendance).

Plan

Évolution de séquence à séquence



Chaînes de dépendance



Encodages

Encodages de nucléotides :

- $\pi(A) := R$; $\pi(G) := R$; $\pi(C) := Y$; $\pi(T) := Y$,
- $\rho(A) := R$; $\rho(G) := R$; $\rho(C) := C$; $\rho(T) := T$,
- $\eta(A) := A$; $\eta(G) := G$; $\eta(C) := Y$; $\eta(T) := Y$.

Φ-encodage d'une séquence de nucléotides :

$$\Phi(x_1(t), \ldots, x_m(t)) = (\rho(x_1(t)), x_2(t), \ldots, x_{m-1}(t), \eta(x_m(t))).$$

Encodages

Encodages de nucléotides :

- $\pi(A) := R$; $\pi(G) := R$; $\pi(C) := Y$; $\pi(T) := Y$,
- $\rho(A) := R$; $\rho(G) := R$; $\rho(C) := C$; $\rho(T) := T$,
- $\eta(A) := A$; $\eta(G) := G$; $\eta(C) := Y$; $\eta(T) := Y$.

Φ-encodage d'une séquence de nucléotides :

$$\Phi(x_1(t), \ldots, x_m(t)) = (\rho(x_1(t)), x_2(t), \ldots, x_{m-1}(t), \eta(x_m(t))).$$

Théorème pour l'évolution Φ-encodée

Évolution Φ-encodée d'une séquence

$$(\rho(X_1)(t), X_2(t), \ldots, X_{m-1}(t), \eta(X_m)(t))_{t\in[0,T]}$$

issue d'un modèle RN95+YpR :

Théorème [BGP08]

- Une séquence Φ-encodée évolue dans le temps selon une chaîne de Markov explicite.
- Des séquences Φ-encodées disjointes évoluent indépendamment.

Intérêt pour le calcul de la vraisemblance

- Calcul de la vraisemblance pour des séquences Φ-encodées de longueurs $m = 2, 3, 4, 5$.
- Découpage RY en séquences Φ-encodées minimales.
  Exemple :

$$L(\Phi(\texttt{TAATGGA})) = L(\Phi(\texttt{TAA})\Phi(\texttt{TGGA})) = L(\Phi(\texttt{TAA}))L(\Phi(\texttt{TGGA})).$$

$\rightarrow$ Calcul de la vraisemblance pour des observations générales ?

- Vraisemblances composites.
- Approximation de type Monte Carlo.

Intérêt pour le calcul de la vraisemblance

- Calcul de la vraisemblance pour des séquences Φ-encodées de longueurs $m = 2, 3, 4, 5$.
- Découpage RY en séquences Φ-encodées minimales.
  Exemple :

$$L(\Phi(\texttt{TAATGGA})) = L(\Phi(\texttt{TAA})\Phi(\texttt{TGGA})) = L(\Phi(\texttt{TAA}))L(\Phi(\texttt{TGGA})).$$

$\rightarrow$ Calcul de la vraisemblance pour des observations générales ?

- Vraisemblances composites.
- Approximation de type Monte Carlo.

Intérêt pour le calcul de la vraisemblance

- Calcul de la vraisemblance pour des séquences Φ-encodées de longueurs $m = 2, 3, 4, 5$.
- Découpage RY en séquences Φ-encodées minimales.
  Exemple :

$$L(\Phi(\texttt{TAATGGA})) = L(\Phi(\texttt{TAA})\Phi(\texttt{TGGA})) = L(\Phi(\texttt{TAA}))L(\Phi(\texttt{TGGA})).$$

$\rightarrow$ Calcul de la vraisemblance pour des observations générales ?

- Vraisemblances composites.
- Approximation de type Monte Carlo.

Vraisemblances composites : triplets encodés

Découpage des séquences en triplets Φ-encodés [BG12] :

$$\begin{array}{ccc} \underbrace{X_1(T)X_2(T)X_3(T)} & \underbrace{X_4(T)X_5(T)X_6(T)} & \dots \\ \Phi(X_1(T)X_2(T)X_3(T)) & \Phi(X_4(T)X_5(T)X_6(T)) & \dots \end{array}$$

On déduit pour chaque paramètre θ une vraisemblance composite $\hat{L}_{\text{triplets}}(\theta)$.

$\rightarrow$ Cette vraisemblance composite a été exploitée numériquement dans [BG12].

Vraisemblances composites : triplets encodés

Découpage des séquences en triplets Φ-encodés [BG12] :

$$\begin{array}{cccc} \underbrace{X_1(T)X_2(T)X_3(T)} & & \underbrace{X_4(T)X_5(T)X_6(T)} & \dots \\ \Phi(X_1(T)X_2(T)X_3(T)) & & \Phi(X_4(T)X_5(T)X_6(T)) & \dots \end{array}$$

On déduit pour chaque paramètre θ une vraisemblance composite $\hat{L}_{\text{triplets}}(\theta)$.

$\rightarrow$ Cette vraisemblance composite a été exploitée numériquement dans [BG12].

Vraisemblances composites : approximation markovienne

- $\hat{L}_{\text{triplets}}(\theta)$ définie à l'aide des triplets Φ-encodés,
- $\hat{L}_{\text{couples}}(\theta)$ définie de même avec les couples Φ-encodés.

Définition

$$\hat{L}_{\text{Markov}}(\theta) = \frac{\hat{L}_{\text{triplets}}(\theta)}{\hat{L}_{\text{couples}}(\theta)}.$$

- Justification du nom : approximation exacte si

$$P(\Phi(X_iX_{i+1})(T)|\Phi(X_1\dots X_i)(T)) = P(\Phi(X_iX_{i+1})(T)|\Phi(X_{i-1}X_i)(T)).$$

- Intérêt : approximation composite de la vraisemblance.

Vraisemblances composites : approximation markovienne

- $\hat{L}_{\text{triplets}}(\theta)$ définie à l'aide des triplets Φ-encodés,
- $\hat{L}_{\text{couples}}(\theta)$ définie de même avec les couples Φ-encodés.

Définition

$$\hat{L}_{\text{Markov}}(\theta) = \frac{\hat{L}_{\text{triplets}}(\theta)}{\hat{L}_{\text{couples}}(\theta)}.$$

- Justification du nom : approximation exacte si

$$P(\Phi(X_iX_{i+1})(T)|\Phi(X_1\dots X_i)(T)) = P(\Phi(X_iX_{i+1})(T)|\Phi(X_{i-1}X_i)(T)).$$

- Intérêt : approximation composite de la vraisemblance.

Plan

Structure de champ markovien d'ordre un

Évolution Φ-encodée d'une séquence issue d'un modèle RN95+YpR :

$$\Phi(X) = (\rho(X_1)(t), X_2(t), \ldots, X_{m-1}(t), \eta(X_m)(t))_{t\in[0,T]}$$

Théorème (thèse H.)

La séquence évolutive Φ-encodée $\{(\Phi(X))_i \; ; \; i \in [\![1, m]\!]\}$ est un champ markovien d'ordre un.

- Forme explicite de l'évolution d'un site i conditionnellement aux autres sites.
- Induit une structure spatiale de chaîne de Markov... Mais forme non explicite.

Structure de champ markovien d'ordre un

Évolution Φ-encodée d'une séquence issue d'un modèle RN95+YpR :

$$\Phi(X) = (\rho(X_1)(t), X_2(t), \ldots, X_{m-1}(t), \eta(X_m)(t))_{t\in[0,T]}$$

Théorème (thèse H.)

La séquence évolutive Φ-encodée $\{(\Phi(X))_i\ ;\ i \in [\![1, m]\!]\}$ est un champ markovien d'ordre un.

- Forme explicite de l'évolution d'un site i conditionnellement aux autres sites.
- Induit une structure spatiale de chaîne de Markov... Mais forme non explicite.

Structure de chaîne de Markov explicite

Plutôt que de regarder l'évolution site par site, on regarde l'évolution de chaque dinucléotide Φ-encodé avec chevauchement.

Définition

$$\rho_i = (\rho(X_i(t)))_t \text{ et } \eta_i = (\eta(X_i(t)))_t,$$

$$Z_i = (\rho_i, \eta_{i+1}).$$

Alphabet associé : $\{C, T, R\} \times \{A, G, Y\}$.

$$\begin{aligned}\Phi(X) &= (\rho_1, X_2, \ldots, X_{m-1}, \eta_m) \\ &\equiv (\rho_1, \eta_2, \rho_2, \ldots, \eta_{m-1}, \rho_{m-1}, \eta_m) \\ &= (Z_1, \ldots, Z_{m-1}).\end{aligned}$$

Structure de chaîne de Markov explicite

Plutôt que de regarder l'évolution site par site, on regarde l'évolution de chaque dinucléotide Φ-encodé avec chevauchement.

Définition

$$\rho_i = (\rho(X_i(t)))_t \text{ et } \eta_i = (\eta(X_i(t)))_t,$$

$$Z_i = (\rho_i, \eta_{i+1}).$$

Alphabet associé : $\{C, T, R\} \times \{A, G, Y\}$.

$$\begin{aligned}\Phi(X) &= (\rho_1, X_2, \ldots, X_{m-1}, \eta_m) \\ &\equiv (\rho_1, \eta_2, \rho_2, \ldots, \eta_{m-1}, \rho_{m-1}, \eta_m) \\ &= (Z_1, \ldots, Z_{m-1}).\end{aligned}$$

Structure de chaîne de Markov explicite

Théorème (thèse H.)

On suppose la racine fixée. Alors $(Z_i)_i$ est une chaîne de Markov.

Théorème (thèse H.)

Conditionnellement à $Z_{1:i-1}$ et $Z_i([0,t[)$, la description de la loi de transition de Z_i à l'instant t est explicite :

1. Si $\pi_i(t^-) = \pi_i(t) \in \{R, Y\}$, matrice de taux de sauts W_R ou W_Y.
2. Si $\pi_i(t^-) \neq \pi_i(t)$, substitution obligatoire régie par des matrices instantanées $U_{Y\to R}$ et $U_{R\to Y}$.

Structure de chaîne de Markov explicite

Théorème (thèse H.)

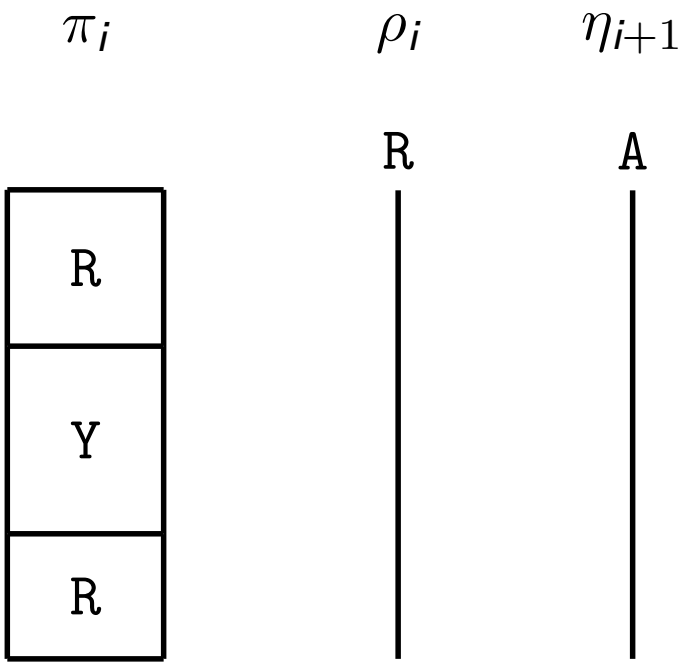
On suppose la racine fixée. Alors $(Z_i)_i$ est une chaîne de Markov.

Théorème (thèse H.)

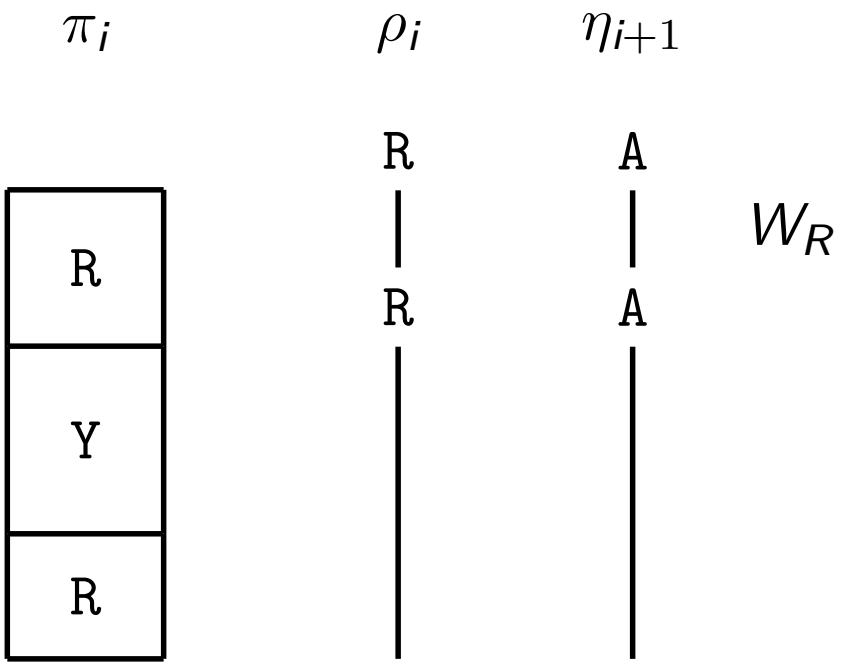
Conditionnellement à $Z_{1:i-1}$ et $Z_i([0,t[)$, la description de la loi de transition de Z_i à l'instant t est explicite :

1. Si $\pi_i(t^-) = \pi_i(t) \in \{R, Y\}$, matrice de taux de sauts W_R ou W_Y.
2. Si $\pi_i(t^-) \neq \pi_i(t)$, substitution obligatoire régie par des matrices instantanées $U_{Y\to R}$ et $U_{R\to Y}$.

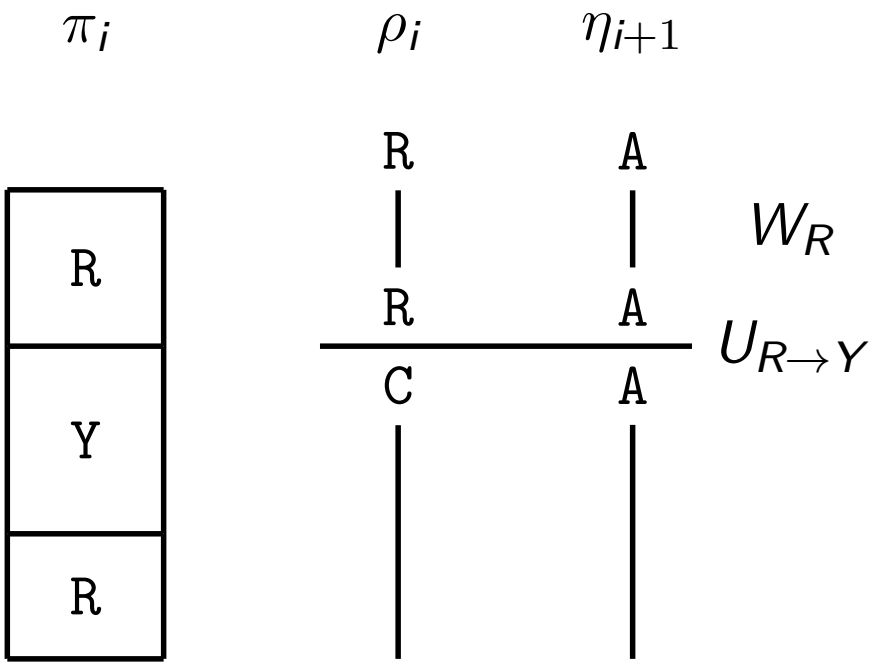
Exemple



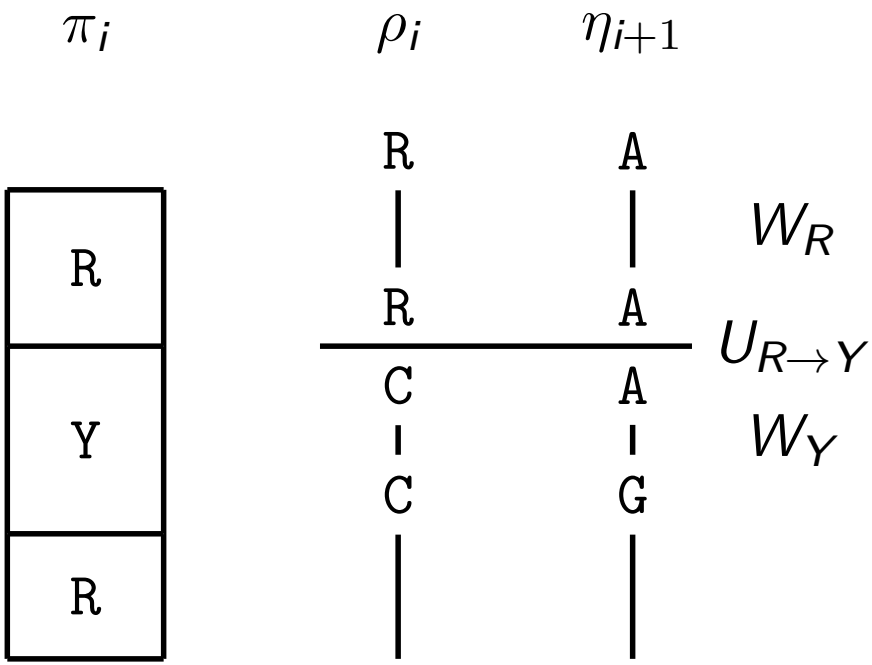
Exemple



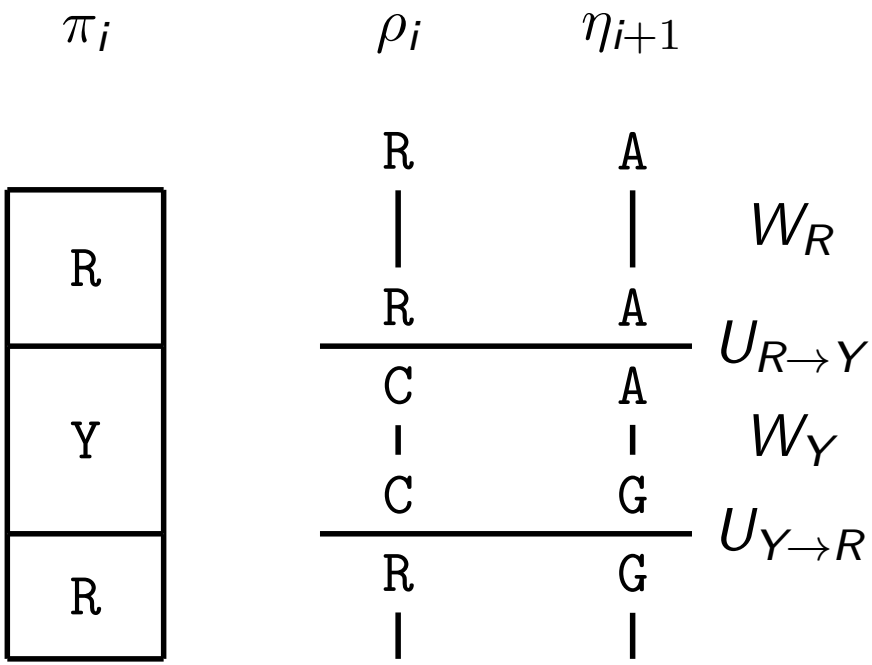
Exemple



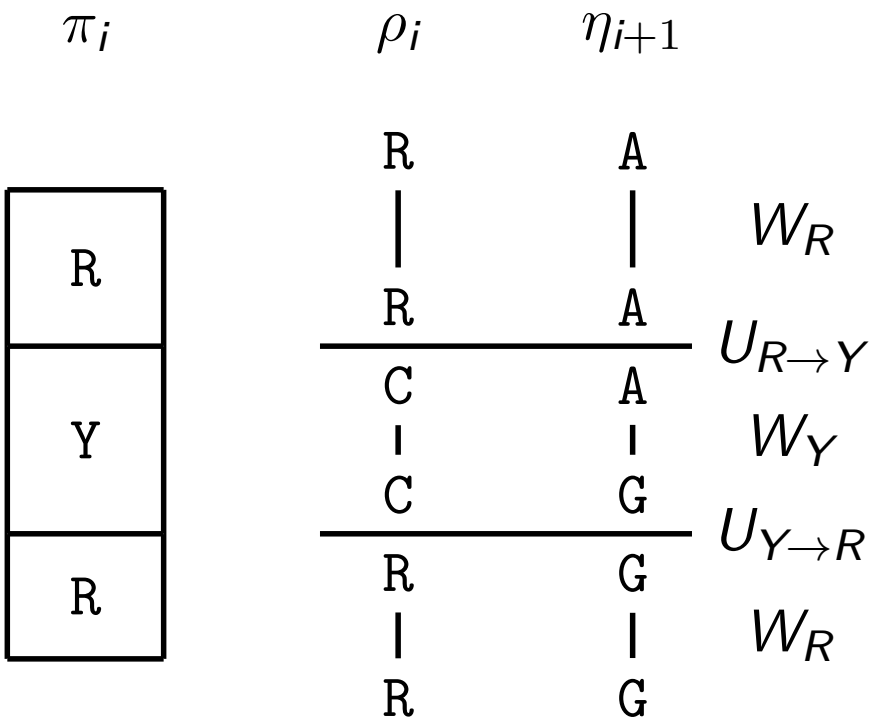
Exemple



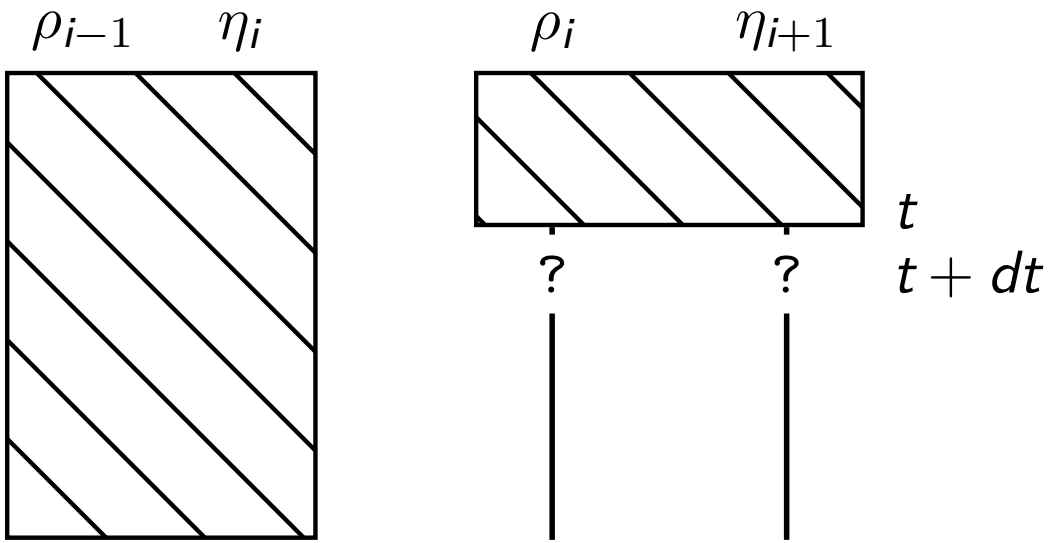
Exemple



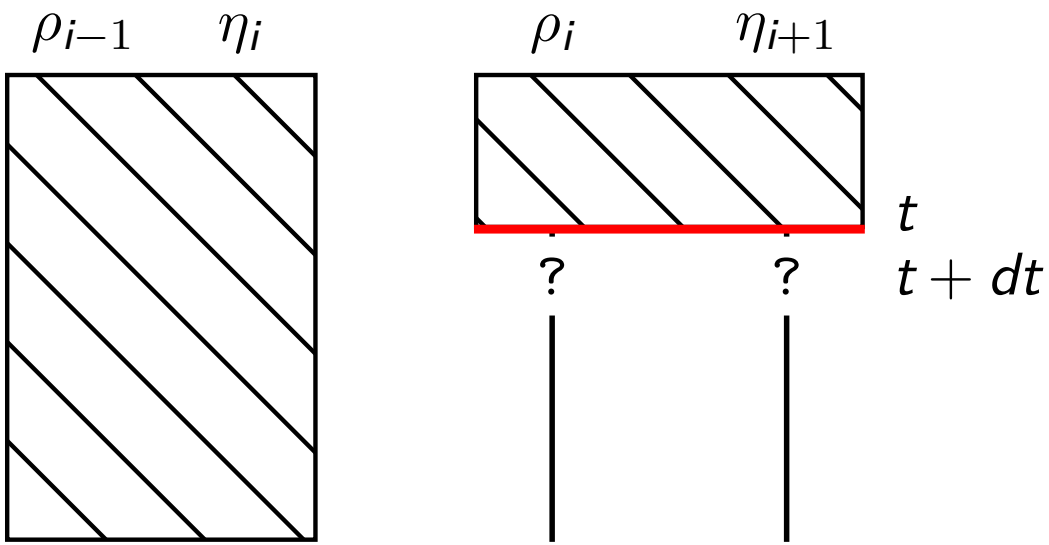
Exemple



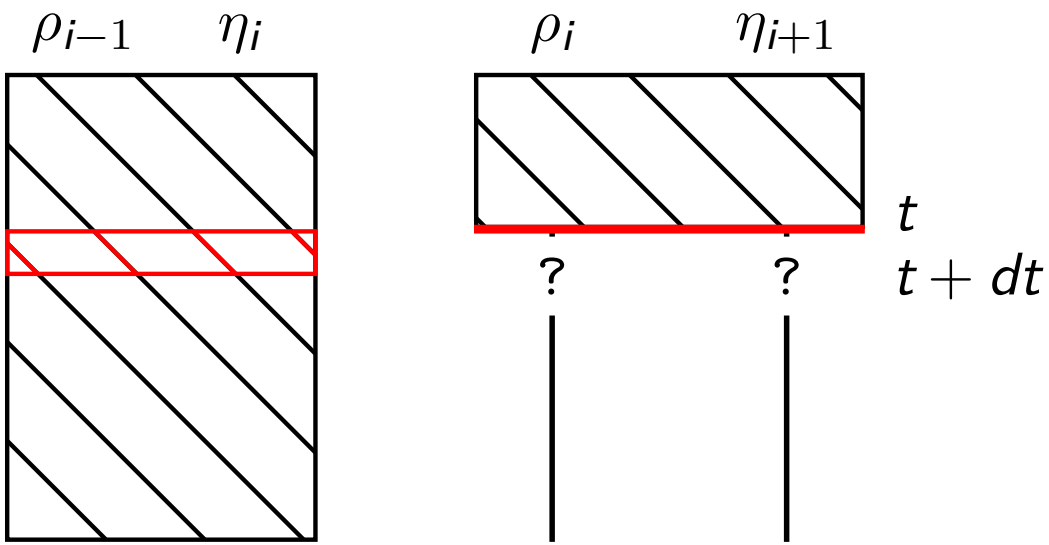
Idée de la démonstration



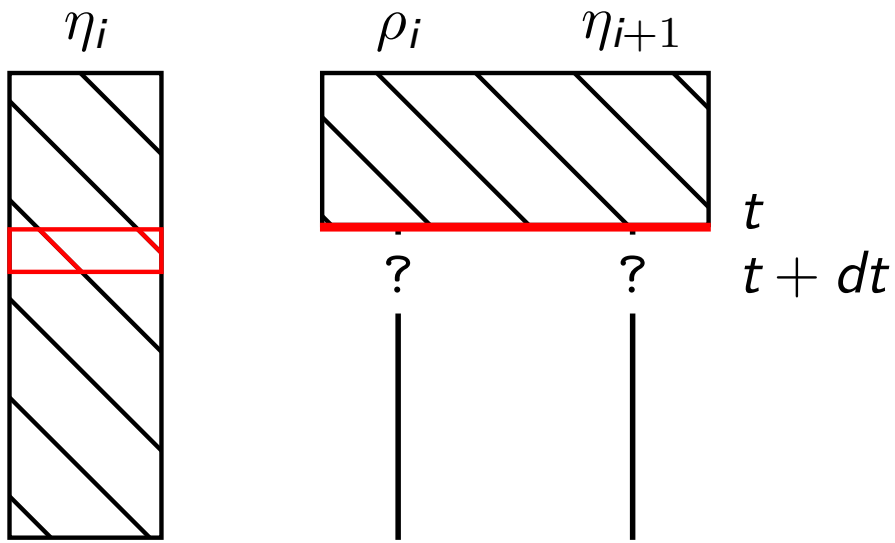
Idée de la démonstration



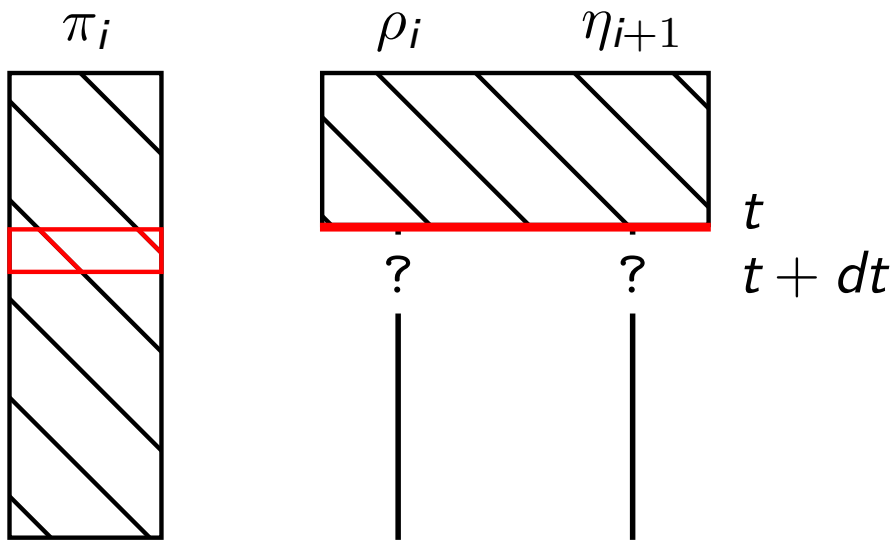
Idée de la démonstration



Idée de la démonstration



Idée de la démonstration



Estimateur du maximum de vraisemblance

Pour l'évolution markovienne de l'historique $(Z_i)_i$, on a :

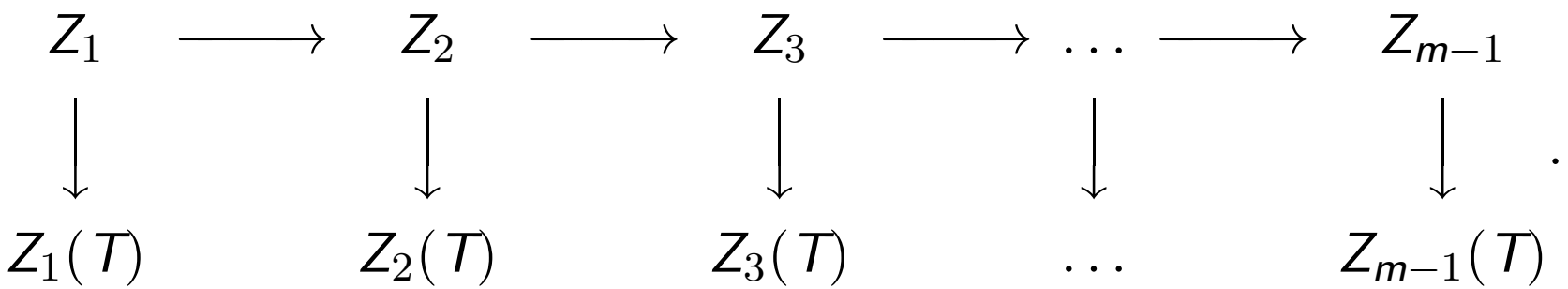
$$\begin{array}{ccccccccc} Z_1 & \longrightarrow & Z_2 & \longrightarrow & Z_3 & \longrightarrow & \dots & \longrightarrow & Z_{m-1} \\ \downarrow & & \downarrow & & \downarrow & & \downarrow & & \downarrow \\ Z_1(T) & & Z_2(T) & & Z_3(T) & & \dots & & Z_{m-1}(T) \end{array} .$$

Théorème (thèse H.)

Consistance et normalité asymptotique de l'estimateur du maximum de vraisemblance pour des observations issues d'un modèle RN95+YpR (sous des conditions de compacité et d'identifiabilité des paramètres).

Estimateur du maximum de vraisemblance

Pour l'évolution markovienne de l'historique $(Z_i)_i$, on a :



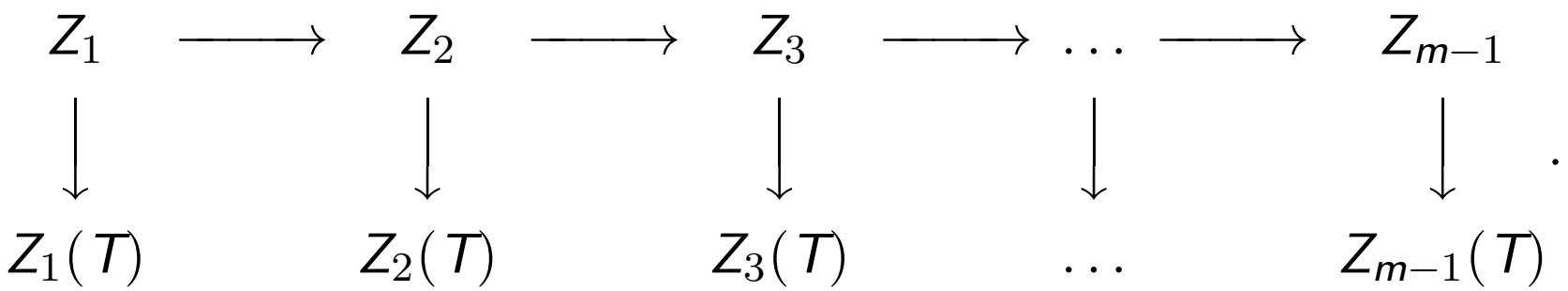
Théorème (thèse H.)

Consistance et normalité asymptotique de l'estimateur du maximum de vraisemblance pour des observations issues d'un modèle RN95+YpR (sous des conditions de compacité et d'identifiabilité des paramètres).

Plan

Comment approcher la vraisemblance ?

Pour l'évolution markovienne de l'historique $(Z_i)_i$, on a :



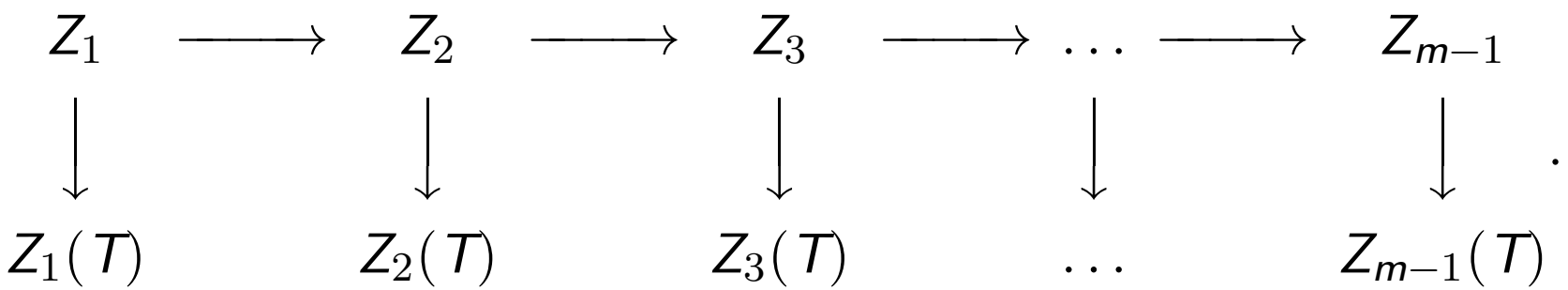
- Connaissance de $z_{1:m-1}(T)$ mais pas de $z_{1:m-1}$.
- But : calculer la vraisemblance des observations $z_{1:m-1}(T)$ par produit :

$$p(z_{1:m-1}(T)) = \prod_{i=0}^{m-2} p(z_{i+1}(T)|z_{1:i}(T)).$$

- Méthode : filtre particulaire auxiliaire (APF).

Comment approcher la vraisemblance ?

Pour l'évolution markovienne de l'historique $(Z_i)_i$, on a :



- Connaissance de $z_{1:m-1}(T)$ mais pas de $z_{1:m-1}$.
- But : calculer la vraisemblance des observations $z_{1:m-1}(T)$ par produit :

$$p(z_{1:m-1}(T)) = \prod_{i=0}^{m-2} p(z_{i+1}(T)|z_{1:i}(T)).$$

- Méthode : filtre particulaire auxiliaire (APF).

Algorithme APF

Idée de l'algorithme : approcher les lois de $z_{1:i}$ conditionnellement à $z_{1:i}(T)$ par la loi empirique associée à un nuage de particules.

- Site $i-1$:

$$\frac{1}{N}\sum_{j=1}^{N} 1_{z_{1:i-1}^{(j)}}(z_{1:i-1}) \approx \text{loi}(z_{1:i-1}|z_{1:i-1}(T)).$$

- Pour toute particule $j \in 1:N$, conditionnellement à $z_{i-1}^{(j)}$, simuler selon $p(dz_i|z_{i-1}, z_i(T))$:

$$z_i^{(1)}, \ldots, z_i^{(N)}.$$

- Site i :

$$\frac{1}{N}\sum_{j=1}^{N} 1_{z_{1:i}^{(j)}}(z_{1:i}) \approx \text{loi}(z_{1:i}|z_{1:i}(T)).$$

Puis :

$$p(z_{i+1}(T)|z_{1:i}(T)) = \int p(z_{i+1}(T)|z_i)p(dz_{1:i}|z_{1:i}(T)).$$

Algorithme APF

Idée de l'algorithme : approcher les lois de $z_{1:i}$ conditionnellement à $z_{1:i}(T)$ par la loi empirique associée à un nuage de particules.

- Site $i-1$:
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i-1}^{(j)}}(z_{1:i-1}) \approx \text{loi}(z_{1:i-1}|z_{1:i-1}(T)).$$

- Pour toute particule $j \in 1:N$, conditionnellement à $z_{i-1}^{(j)}$, simuler selon $p(dz_i|z_{i-1}, z_i(T))$:
$$z_i^{(1)}, \ldots, z_i^{(N)}.$$

- Site i :
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i}^{(j)}}(z_{1:i}) \approx \text{loi}(z_{1:i}|z_{1:i}(T)).$$

Puis :
$$p(z_{i+1}(T)|z_{1:i}(T)) = \int p(z_{i+1}(T)|z_i)p(dz_{1:i}|z_{1:i}(T)).$$

Algorithme APF

Idée de l'algorithme : approcher les lois de $z_{1:i}$ conditionnellement à $z_{1:i}(T)$ par la loi empirique associée à un nuage de particules.

- Site $i-1$:
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i-1}^{(j)}}(z_{1:i-1}) \approx \text{loi}(z_{1:i-1}|z_{1:i-1}(T)).$$
- Pour toute particule $j \in 1:N$, conditionnellement à $z_{i-1}^{(j)}$, simuler selon $p(dz_i|z_{i-1}, z_i(T))$:
$$z_i^{(1)}, \ldots, z_i^{(N)}.$$
- Site i :
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i}^{(j)}}(z_{1:i}) \approx \text{loi}(z_{1:i}|z_{1:i}(T)).$$

Puis :
$$p(z_{i+1}(T)|z_{1:i}(T)) = \int p(z_{i+1}(T)|z_i)p(dz_{1:i}|z_{1:i}(T)).$$

Algorithme APF

Idée de l'algorithme : approcher les lois de $z_{1:i}$ conditionnellement à $z_{1:i}(T)$ par la loi empirique associée à un nuage de particules.

- Site $i-1$:
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i-1}^{(j)}}(z_{1:i-1}) \approx \text{loi}(z_{1:i-1}|z_{1:i-1}(T)).$$

- Pour toute particule $j \in 1:N$, conditionnellement à $z_{i-1}^{(j)}$, simuler selon $p(dz_i|z_{i-1}, z_i(T))$:
$$z_i^{(1)}, \dots, z_i^{(N)}.$$

- Site i :
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i}^{(j)}}(z_{1:i}) \approx \text{loi}(z_{1:i}|z_{1:i}(T)).$$

Puis :

$$p(z_{i+1}(T)|z_{1:i}(T)) = \int p(z_{i+1}(T)|z_i)p(dz_{1:i}|z_{1:i}(T)).$$

Algorithme APF

Idée de l'algorithme : approcher les lois de $z_{1:i}$ conditionnellement à $z_{1:i}(T)$ par la loi empirique associée à un nuage de particules.

- Site $i-1$:
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i-1}^{(j)}}(z_{1:i-1}) \approx \text{loi}(z_{1:i-1}|z_{1:i-1}(T)).$$

- Pour toute particule $j \in 1:N$, conditionnellement à $z_{i-1}^{(j)}$, simuler selon $p(dz_i|z_{i-1}, z_i(T))$:
$$z_i^{(1)}, \dots, z_i^{(N)}.$$

- Site i :
$$\frac{1}{N}\sum_{j=1}^{N} \mathbf{1}_{z_{1:i}^{(j)}}(z_{1:i}) \approx \text{loi}(z_{1:i}|z_{1:i}(T)).$$

Puis :
$$p(z_{i+1}(T)|z_{1:i}(T)) = \int p(z_{i+1}(T)|z_i)p(dz_{1:i}|z_{1:i}(T)).$$

Convergence (APF)

Théorème

Pour un nombre de sites m fixé, l'estimateur de $p(z_{i+1}(T)|z_{1:i}(T))$ est consistant et asymptotiquement normal lorsque le nombre de particules N tend vers l'infini.

Théorème

Pour un nombre de sites m fixé, le produit des estimateurs de $p(z_{i+1}(T)|z_{1:i}(T))$ (pour $i \in 0 : m-2$) est un estimateur consistant et asymptotiquement normal de $p(z_{1:m-1}(T))$ lorsque le nombre de particules N tend vers l'infini.

Implémentation

Entrées (sous forme de fichier texte) :

- le jeu de séquences, arbre, paramètres du modèle
- la loi à la racine (modèle markovien),
- N le nombre de particules utilisées.

Sortie :

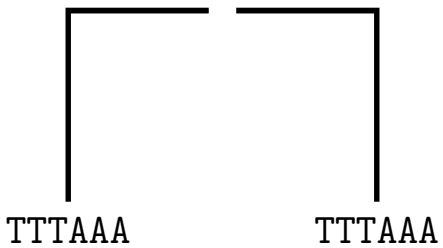
- la log-vraisemblance approchée du jeu de séquences observé.

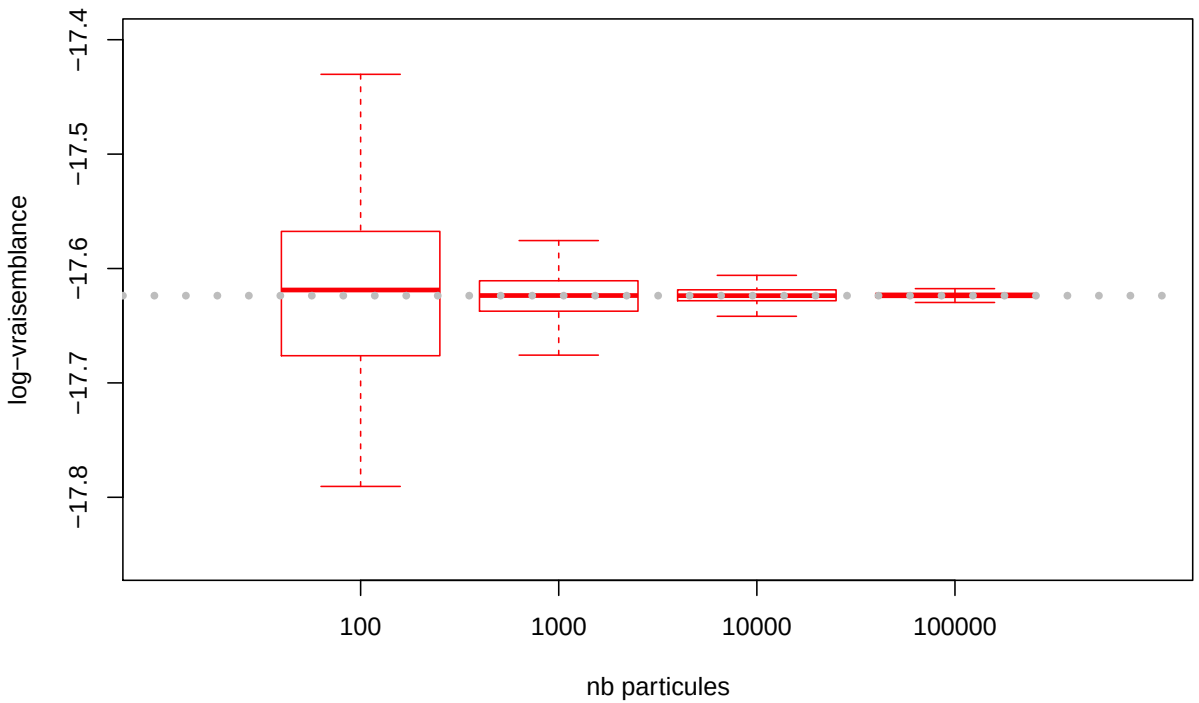
Implémentation

```
class EvolSite
{
 public:
//////////////////
// Constructeurs //
//////////////////
 EvolSite(int lettreInitialeAlph, std::vector<double>
tempsChangementGauche,  std::vector<char> quelChangementGauche, double
tempsInit, double tempsFinal, Noyau &noyau); //constructeur complet,
classique.
 EvolSite(std::vector<double> tempsChangementGauche,  std::vector<char>
quelChangementGauche, double tempsInit, double tempsFinal, Noyau
&noyau); //sans lettre initiale
 EvolSite(int lettreInitialeAlph, double tempsInit, double tempsFinal,
Noyau &noyau); //sans renseignement de ce qui est à gauche
 EvolSite(double tempsInit, double tempsFinal, Noyau &noyau); //sans
connaissance à gauche ni de la lettre initiale.
 EvolSite();

////////////////////////////////////////////
// Méthodes constantes pour le site en cours //
////////////////////////////////////////////
//temps
 double getTempsInit() const;
 double getTempsFinal() const;
 double getTempsActuel() const;
//dernier changement
 int getQuelChangementMatBack() const; //parmi 0..2 ou 0..5
 int getQuelChangementAlphBack() const; //parmi 0..8
//nucléotide à la racine
 bool racineEstVide() const; //1 si pas de site initial fixé. 0 non vide.
1 vide.
 int getLettreInitialeMat() const; //parmi 0..2 ou 0..5
 int getLettreInitialeAlph() const; //parmi 0..8
//noyau
 Noyau& getNoyau() const; //sert pour l'évolution sur l'arbre pour
obtenir les différents taux

//////////////////////////////////////////////////////////////
// Méthodes constantes pour le site à gauche du site évoluant //
//////////////////////////////////////////////////////////////
//au temps t
 char elementAGaucheC(double t) const; //élément à gauche parmi 'R' ou
'Y' lorsqu'on est au temps t
 int indiceProchainChangementAGauche(double t) const; //indice du vecteur
du prochain changement à gauche après t (strictement)
 double tempsProchainChangementAGauche(double t) const; //instant du
prochain changement à gauche après t (strictement)
//au temps final
 char getQuelChangementGaucheFinal() const;
//écarts entre les différents instants de changement à gauche dans {R,Y},
sert entre autres à obtenir la matrice d'évolution
 std::vector<double> differentsEcartsEntre(double s, double t) const; //
donne les écarts à gauche sur ]s,t]
 std::vector<char> differentsCaractEntre(double s, double t) const; //
donne les caract à gauche sur ]s,t]
//temps passés en Y et en R, sert pour calculer la fonction de survie
 double tempsPasseY(double s, double t) const; //cumul des temps entre s
et t où le pi à gauche du site considéré vaut Y.
 double tempsPasseR(double s, double t) const; //pareil avec R
 double tempsPasse(char N, double s, double t) const; //N parmi {R,Y}.

////////////////////////////////////////////
// Méthodes constantes pour le site évoluant //
////////////////////////////////////////////
//réduit l'information de couples quotientés par la surjection dans
l'ensemble {R,Y}
//Ex: si on a "RA" suivi de "CA" suivi de "CG" suivi de "CY" au site en
cours, alors l'évolution dans pi du nucléotide à droite est 'R' suivi de
'R' suivi de 'R' suivi de 'Y'.
 std::vector<int> indiceChangementRY() const; //prend que les indices des
changements qui font vraiment aller de R vers Y ou le contraire.
 std::vector<double> tempsChangementRY() const;
 std::vector<char> quelChangementRY() const; //dans l'exemple, 'R' puis
'Y'

////////////////////////////////////
// Calcul de la matrice d'évolution // (à partir de l'histoire à gauche
dans {R,Y})
////////////////////////////////////
 std::vector<Eigen::MatrixXd> matriceEvol(double s, double t) const; //
donne les matrices d'évolutions du temps s au temps t, une par inter-
durée.
 void remplirMorceauxExp(); //remplit m_morceauxExp à partir de
matriceEvol, qui sont les différents morceaux que l'on a à multiplier du
temps initial au temps final.
 Eigen::MatrixXd matriceDEvolution(double s) const; //matrices
d'évolution de s jusqu'au temps final (de l'arête). Correspond au produit
des matrices d'évolutions intermédiaires exp(t_1 W_R) * U_{R --> Y} * exp
(t_2 W_Y) etc., où les t_i sont les écarts de temps.
```

Exemple

- Modèle d'évolution $v_A = 7.1, \ \ldots \ r_{TG \to CG} = 7.7.$
- Arbre constitué de deux arêtes de longueur 1.
- Loi à la racine loi stationnaire du modèle.
- Séquences observées : $(TTTAAA, TTTAAA)$.





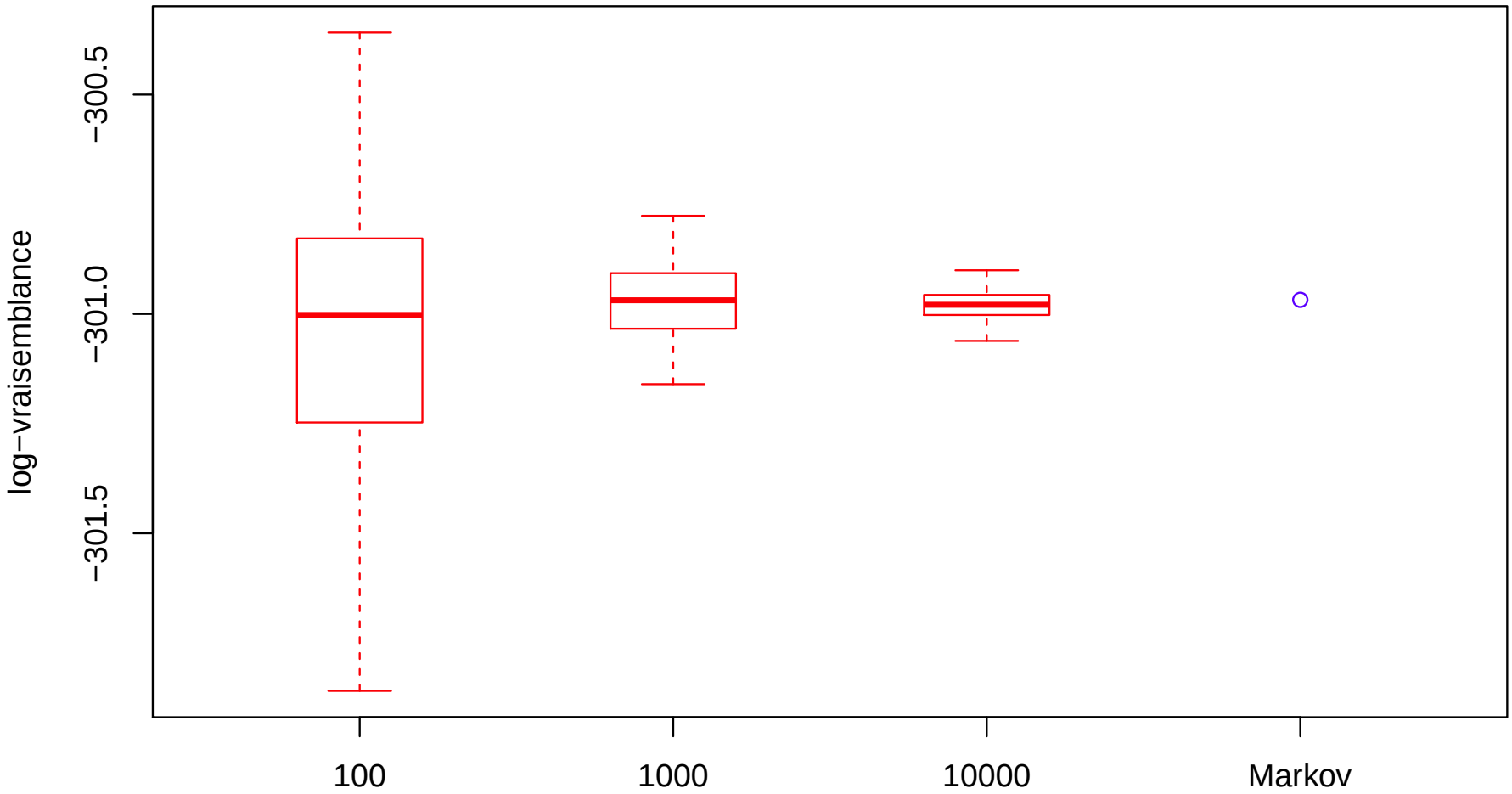
Plan

Comparaisons approx. particulaires / markoviennes

Qualité de l'approximation markovienne ?

Qualité de l'approximation markovienne ?

- Modèle d'évolution et séquences observées tirés aléatoirement, $m = 100$ sites.



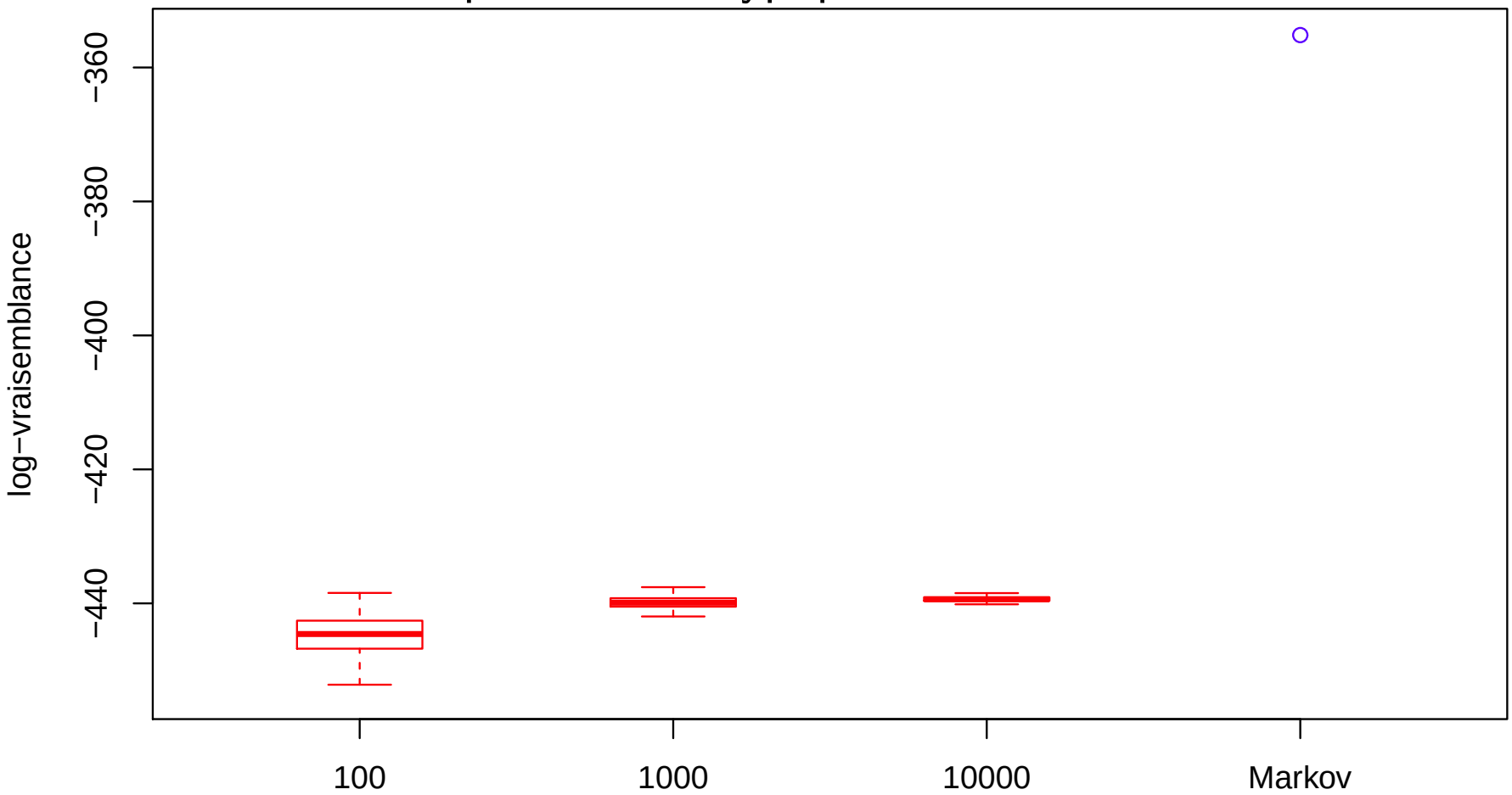
Comparaisons approx. particulaires / markoviennes

Qualité de l'approximation markovienne ?

- Modèle d'évolution et séquences observées choisis pour obtenir un comportement atypique, $m = 100$ sites.

Qualité de l'approximation markovienne ?

- Modèle d'évolution et séquences observées choisis pour obtenir un comportement atypique, $m = 100$ sites.



Inférence d'un nucléotide à la racine

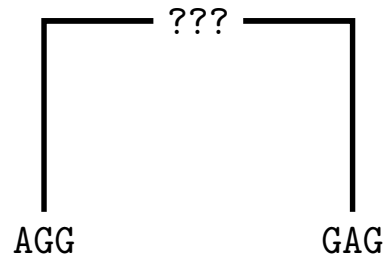
Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Inférence d'un nucléotide à la racine

Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Exemple :

Modèle d'évolution atypique, loi à la racine i.i.d.

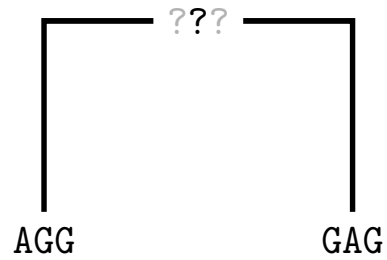


Inférence d'un nucléotide à la racine

Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Exemple :

Modèle d'évolution atypique, loi à la racine i.i.d.

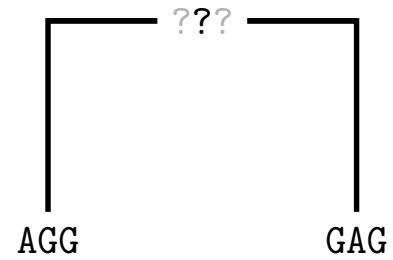


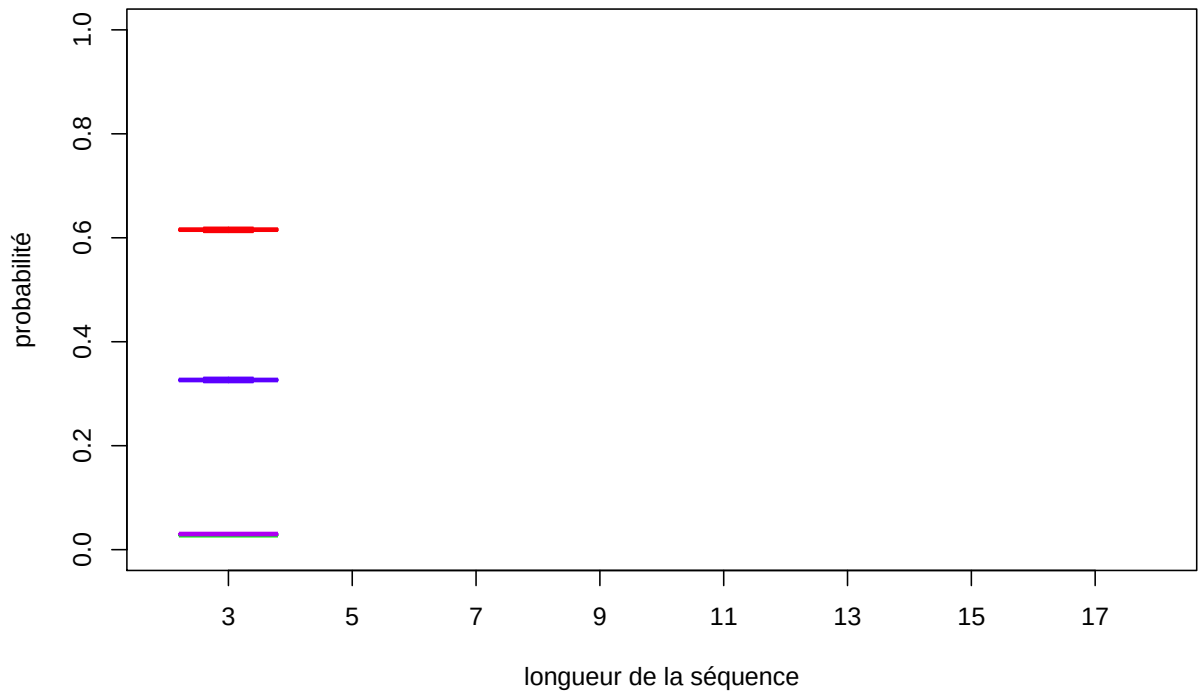
Inférence d'un nucléotide à la racine

Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Exemple :

Modèle d'évolution atypique, loi à la racine i.i.d. A, G, C, T.



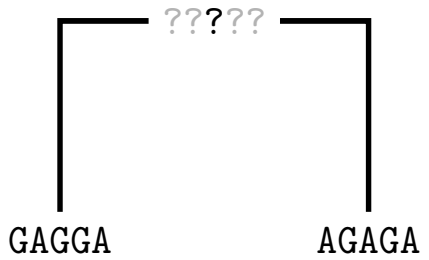


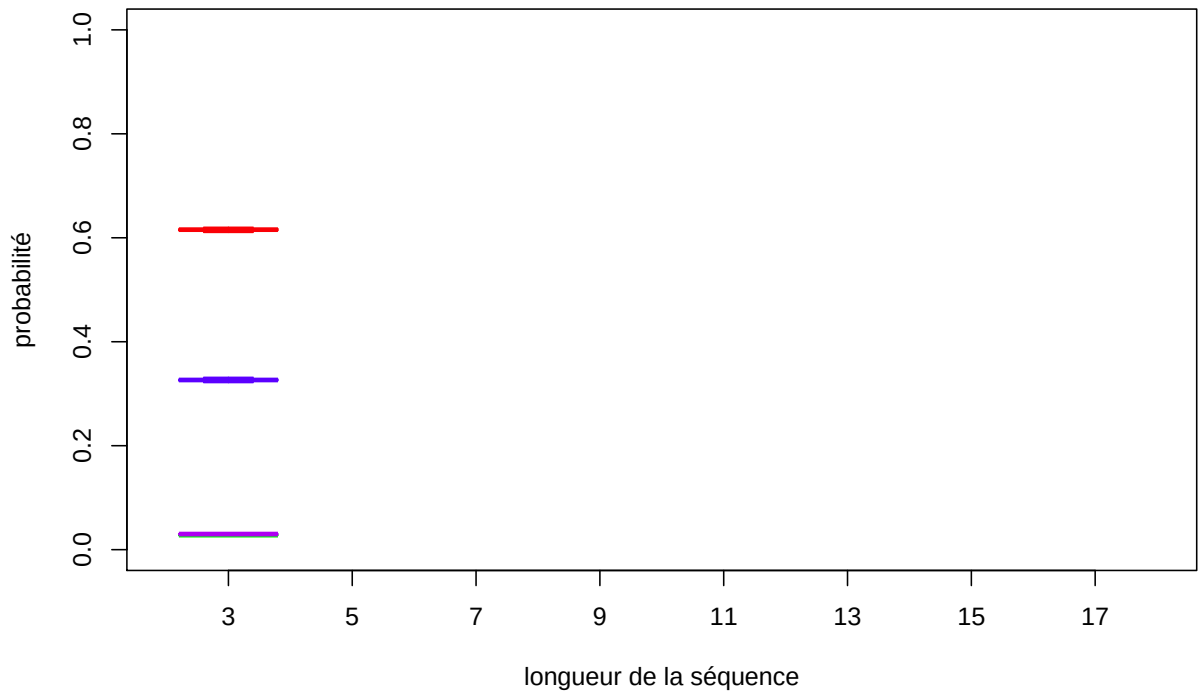
Inférence d'un nucléotide à la racine

Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Exemple :

Modèle d'évolution atypique, loi à la racine i.i.d. A, G, C, T.



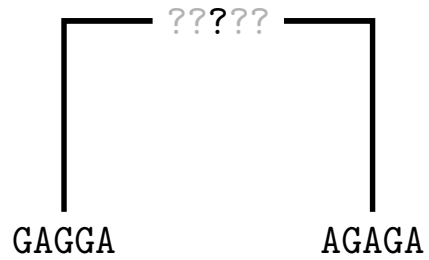


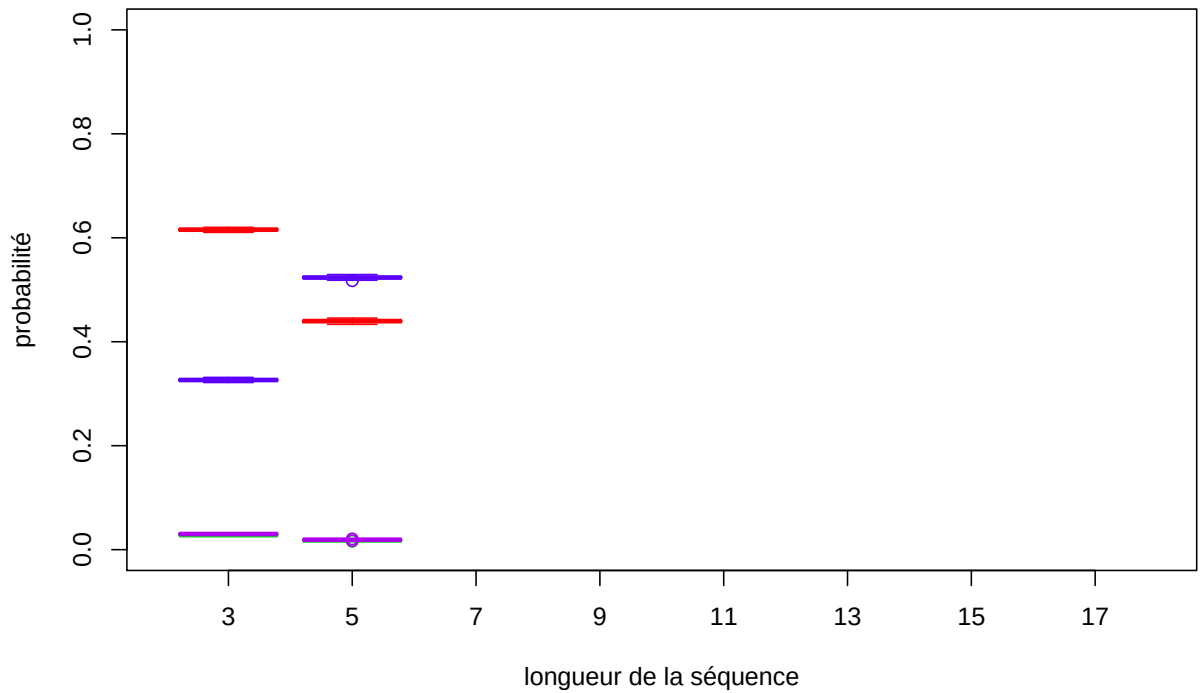
Inférence d'un nucléotide à la racine

Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Exemple :

Modèle d'évolution atypique, loi à la racine i.i.d. A, G, C, T.



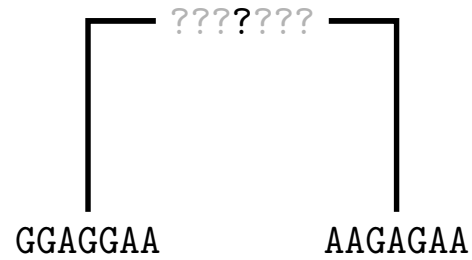


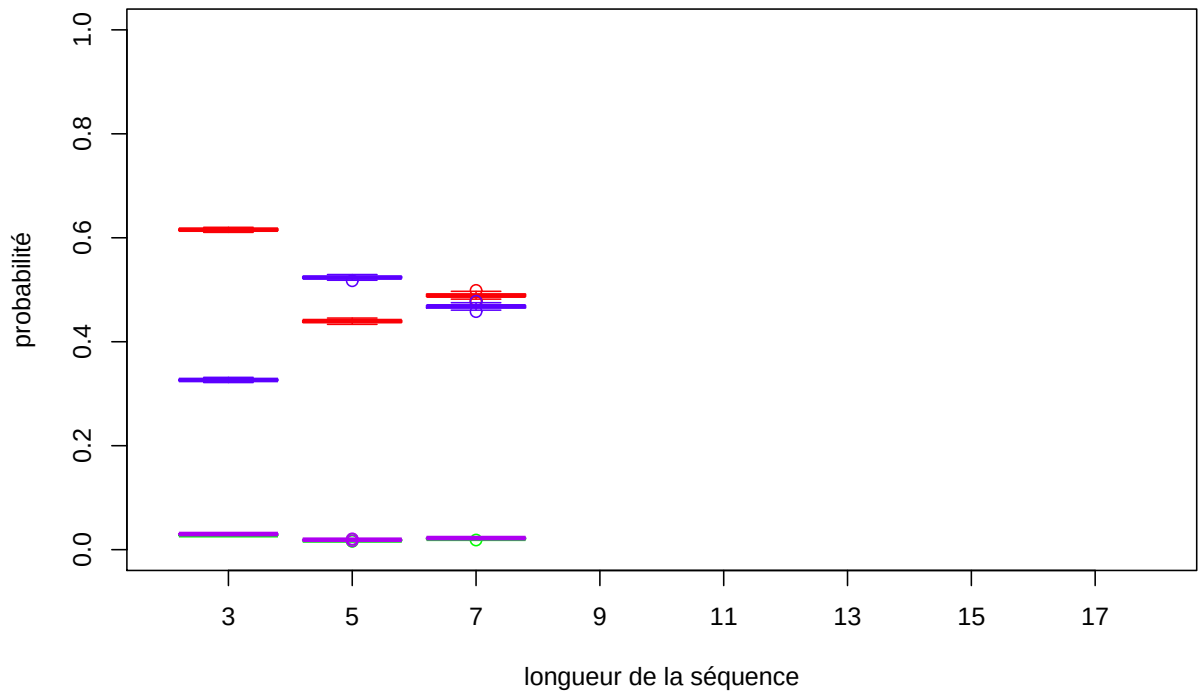
Inférence d'un nucléotide à la racine

Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Exemple :

Modèle d'évolution atypique, loi à la racine i.i.d. A, G, C, T.



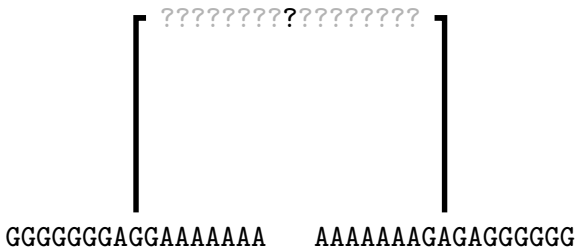


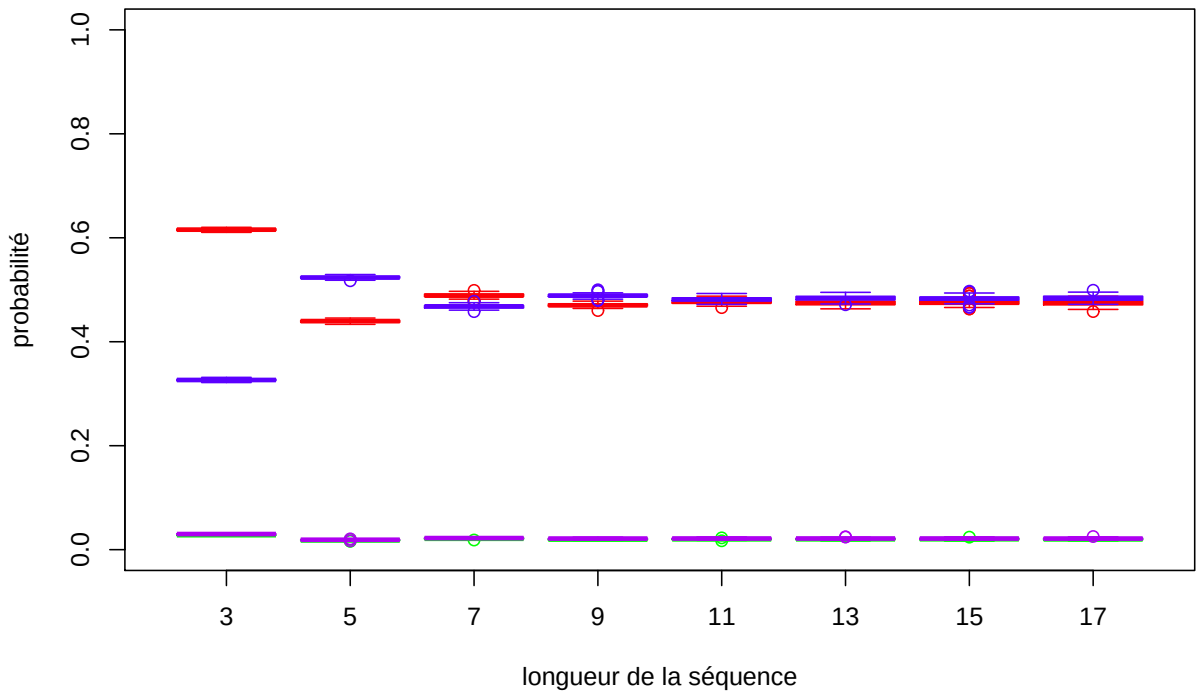
Inférence d'un nucléotide à la racine

Problématique : dépendance aux voisins de l'estimation d'un nucléotide de la séquence ancestrale.

Exemple :

Modèle d'évolution atypique, loi à la racine i.i.d. A, G, C, T.





Comparaison des estimations du max. de vrais.

Comparaison des estimations entre celles obtenues :

- par vraisemblance composite par triplets Φ-encodés,
- par approximations particulaires.

$\rightarrow$ Sur les exemples testés (un paramètre réel à estimer), écart d'estimation « faible ».

$\rightarrow$ Conforte l'utilisation des triplets Φ-encodés pour estimer le max. de vraisemblance.

Comparaison des estimations du max. de vrais.

Comparaison des estimations entre celles obtenues :

- par vraisemblance composite par triplets Φ-encodés,
- par approximations particulaires.

$\rightarrow$ Sur les exemples testés (un paramètre réel à estimer), écart d'estimation « faible ».

$\rightarrow$ Conforte l'utilisation des triplets Φ-encodés pour estimer le max. de vraisemblance.

Comparaison des estimations du max. de vrais.

Comparaison des estimations entre celles obtenues :

- par vraisemblance composite par triplets Φ-encodés,
- par approximations particulaires.

$\rightarrow$ Sur les exemples testés (un paramètre réel à estimer), écart d'estimation « faible ».
$\rightarrow$ Conforte l'utilisation des triplets Φ-encodés pour estimer le max. de vraisemblance.

Algorithme de calcul approché de la vraisemblance

Algorithme pratique

1. Estimer le maximum de vraisemblance des observations par la méthode des triplets encodés (`bppml` de Bio++ [BG12, DB08, DGB+06]).
2. Découper les observations en morceaux indépendants (découpage RY).
3. Pour chaque morceau :
   - Si le nombre de nucléotides est « petit » (< 6), calculer exactement la vraisemblance de ce morceau.
   - Sinon, calculer une approximation particulaire de la vraisemblance du morceau.
4. En déduire une approximation de la vraisemblance au maximum de vraisemblance.

Comparaison de modèles

Exemple d'application : on dispose d'un alignement de trois séquences biologiques de 2215 nucléotides.

modèle	T92	T92+CpGs	GTR
nombre de paramètres	2	3	8
log-vrais.	-3432	?	-3428
AIC	6868	?	6872
BIC	6879	?	6918

$\rightarrow$ Algorithme pratique appliqué pour le modèle T92+CpGs.

Comparaison de modèles

Exemple d'application : on dispose d'un alignement de trois séquences biologiques de 2215 nucléotides.

modèle	T92	T92+CpGs	GTR
nombre de paramètres	2	3	8
log-vrais.	-3432	-3389	-3428
AIC	6868	6784	6872
BIC	6879	6801	6918

$\rightarrow$ Algorithme pratique appliqué pour le modèle T92+CpGs.

Conclusion

$\rightarrow$ Étude d'un modèle d'évolution de l'ADN : RN95+YpR.

$\rightarrow$ Deux manières d'approcher la vraisemblance pour des séquences issues de ces modèles :

- approximations markoviennes,
- approximations particulaires.

$\rightarrow$ Mise en œuvre et comparaison de ces deux approches.

$\rightarrow$ Applications.

Merci de votre attention !