Un modèle de regroupement local d'haplotypes : Browning and Browning (2007).

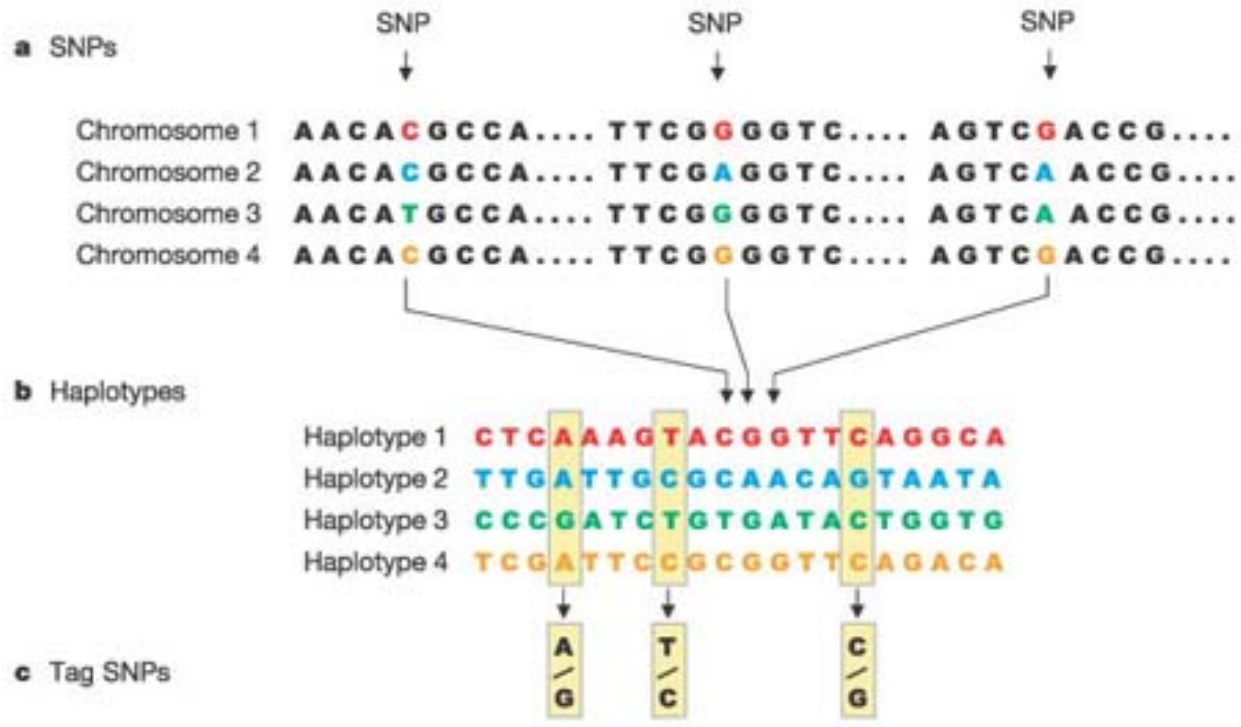
21 juin 2011

Introduction

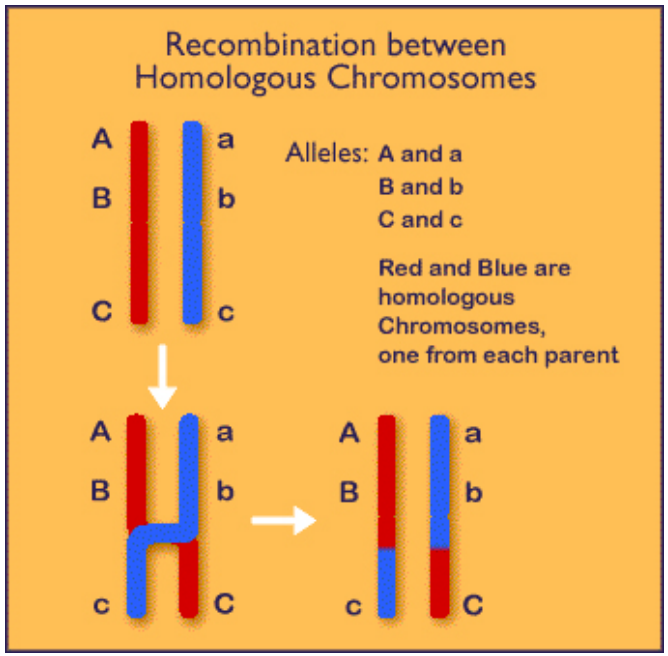
On veut construire des clusters d'haplotypes locaux.
Ça nous permet de :

- modéliser le déséquilibre de liaison (LD) entre marqueurs.
- faire des tests d'association.
- phaser des haplotypes
- prédire les donnés manquantes

Plan de l'exposé



Haplotype : Un échantillon de marqueurs génétiques liées présents dans un même chromosome habituellement transmis ensemble.



Parfois les allèles se transmettent ensemble (AB, ab) et parfois il y a recombinaison (bC,Bc)
$\Rightarrow$ La ségrégation d'allèles adjacents n'est pas indépendante.

Déséquilibre de liaison

$$D_{BC} = p_{BC} - p_B \cdot p_C = p_{BC}p_{bc} - p_{Bc}p_{bC}$$

Chaines de Markov de Longueur Variable non homogènes

Définition

Chaines de Markov dont l'ordre (ou mémoire) dépend de la position et de l'état.

Par exemple, pour une VLMC avec deux états :

$$\mathbb{P}(X_t = x_t | X_{t-1} = 1, X_{t-2}, X_{t-3}, \dots) = \mathbb{P}(X_t = x_t | X_{t-1} = 1)$$

$$\mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2}, X_{t-3}, \dots) = \mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2})$$

où

$$\mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2} = 1) \neq \mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2} = 2)$$

Chaines de Markov de Longueur Variable non homogènes

Définition

Chaines de Markov dont l'ordre (ou mémoire) dépend de la position et de l'état.

Par exemple, pour une VLMC avec deux états :

$$\mathbb{P}(X_t = x_t | X_{t-1} = 1, X_{t-2}, X_{t-3}, \dots) = \mathbb{P}(X_t = x_t | X_{t-1} = 1)$$

$$\mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2}, X_{t-3}, \dots) = \mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2})$$

où

$$\mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2} = 1) \neq \mathbb{P}(X_t = x_t | X_{t-1} = 2, X_{t-2} = 2)$$

Avantages des VLMCs

Déséquilibre de liaison élevé $\Rightarrow$ mémoire longue

Déséquilibre de liaison faible $\Rightarrow$ mémoire courte

Avantage par rapport aux autres méthodes : on n'est pas obligé de choisir une longueur de fenêtre

- Pour les HMMs on a besoin de pre-spécifier la structure du modèle
- Les VLMCs ne nécessitent pas une modélisation spécifique et sont suffisamment flexibles pour s'approcher des HMMs
- On peut les estimer en utilisant des méthodes heuristiques (plus rapides)

Avantages des VLMCs

Déséquilibre de liaison élevé $\Rightarrow$ mémoire longue

Déséquilibre de liaison faible $\Rightarrow$ mémoire courte

Avantage par rapport aux autres méthodes : on n'est pas obligé de choisir une longueur de fenêtre

- Pour les HMMs on a besoin de pre-spécifier la structure du modèle
- Les VLMCs ne nécessitent pas une modélisation spécifique et sont suffisamment flexibles pour s'approcher des HMMs
- On peut les estimer en utilisant des méthodes heuristiques (plus rapides)

Avantages des VLMCs

Déséquilibre de liaison élevé $\Rightarrow$ mémoire longue

Déséquilibre de liaison faible $\Rightarrow$ mémoire courte

Avantage par rapport aux autres méthodes : on n'est pas obligé de choisir une longueur de fenêtre

- Pour les HMMs on a besoin de pre-spécifier la structure du modèle
- Les VLMCs ne nécessitent pas une modélisation spécifique et sont suffisamment flexibles pour s'approcher des HMMs
- On peut les estimer en utilisant des méthodes heuristiques (plus rapides)

Avantages des VLMCs

Déséquilibre de liaison élevé $\Rightarrow$ mémoire longue

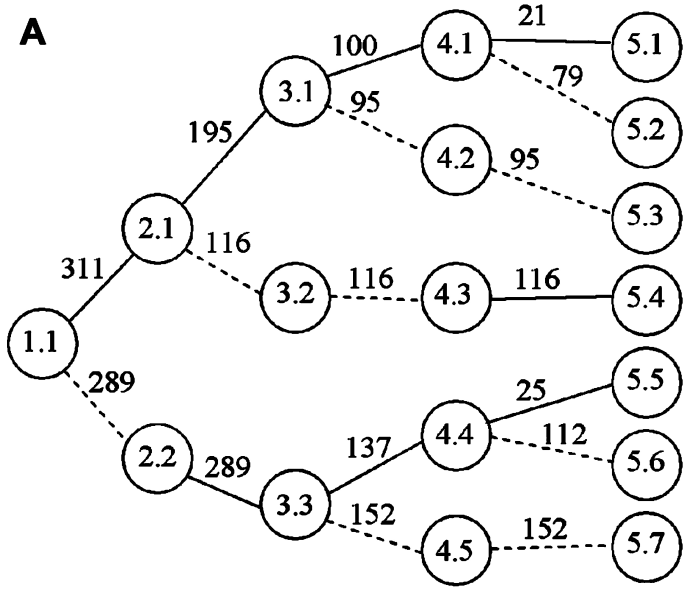
Déséquilibre de liaison faible $\Rightarrow$ mémoire courte

Avantage par rapport aux autres méthodes : on n'est pas obligé de choisir une longueur de fenêtre

- Pour les HMMs on a besoin de pre-spécifier la structure du modèle
- Les VLMCs ne nécessitent pas une modélisation spécifique et sont suffisamment flexibles pour s'approcher des HMMs
- On peut les estimer en utilisant des méthodes heuristiques (plus rapides)

Algorithme pour estimer une VLMC

Une VLMC non homogène peut être représentée par un graphe orienté acyclique (DAG)



Haplotype	Total
1111	21
1112	79
1122	95
1221	116
2111	25
2112	112
2122	152

Figure: Les arrêtes solides entre les niveaux i et $i+1$ représentent l'allèle 1 au SNP i, les pointillés le 0

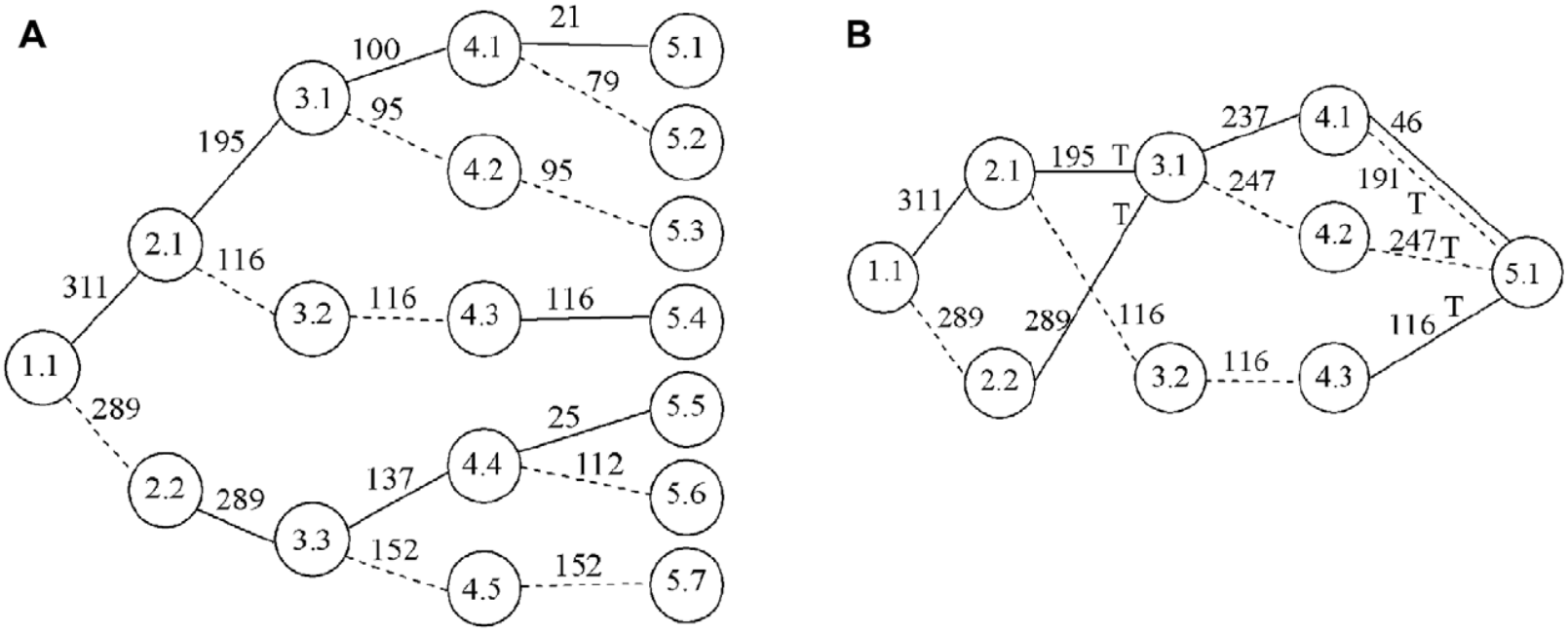


FIGURE: DAG avant et après avoir fini l'algorithme. Les arrêtes représentent des clusters d'haplotypes

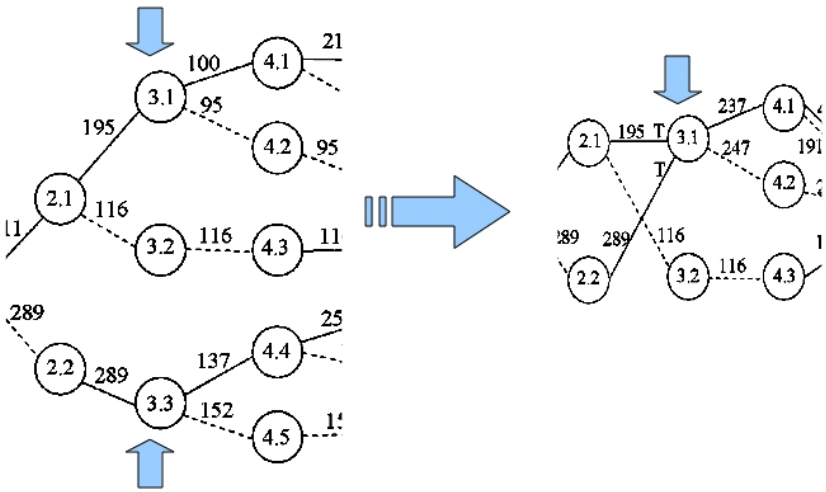
Notation

Chaque niveaux d $(d = 2, \ldots, D+1)$ correspond à la position d'un SNP dans la séquence.

Au niveau 1 il y a seulement un noeud, qui ne contient pas d'information.

Les niveaux $d = 2, \ldots, D+1$ représentent la collection des haplotypes possibles jusqu'au marqueur $d-1$

Une arrête marquée a *qui part d'un noeud* x *au niveau* d enrichit la collection d'haplotypes x de l'allèle a au marqueur d



Fusionner deux noeuds représente l'union de leurs histoires, donc :

$\Rightarrow$ Recombinaison historique

$\Rightarrow$ perte de mémoire dans la chaîne de Markov

On va fusionner deux noeuds si les probas des états futurs sont suffisamment proches

Score de similarité

Notation

n_x : nombre d'haplotypes passant par le noeud x

$n_x(a_d)$: nombre d'haplotypes passant par le noeud x et qui ont l'allèle a_d au marqueur d.

$n_x(a_d a_{d+1})$: nombre d'haplotypes passant par le noeud x et qui ont l'allèle a_d au marqueur d et l'allèle a_{d+1} au marqueur $d+1$.

La différence des probabilités conditionnelles observées pour la séquence $a_d a_{d+1} \dots a_{d+k}$ est

$$diff_{xy}(a_d a_{d+1} \dots a_{d+k}) = \left| \frac{n_x(a_d a_{d+1} \dots a_{d+k})}{n_x} - \frac{n_y(a_d a_{d+1} \dots a_{d+k})}{n_y} \right|$$

Score de similarité

Notation

n_x : nombre d'haplotypes passant par le noeud x

$n_x(a_d)$: nombre d'haplotypes passant par le noeud x et qui ont l'allèle a_d au marqueur d.

$n_x(a_d a_{d+1})$: nombre d'haplotypes passant par le noeud x et qui ont l'allèle a_d au marqueur d et l'allèle a_{d+1} au marqueur $d+1$.

La différence des probabilités conditionnelles observées pour la séquence $a_d a_{d+1} \dots a_{d+k}$ est

$$diff_{xy}(a_d a_{d+1} \dots a_{d+k}) = \left| \frac{n_x(a_d a_{d+1} \dots a_{d+k})}{n_x} - \frac{n_y(a_d a_{d+1} \dots a_{d+k})}{n_y} \right|$$

Score de similarité entre x et y

$$ss(x,y) = \max_{k=0,1,\ldots,D-d} \left(\max_{a_d a_{d+1} \ldots a_{d+k}} \{diff_{xy}(a_d a_{d+1} \ldots a_{d+k})\} \right\}$$

Si $ss(x,y) < \alpha \Rightarrow$ on fusionne les noeuds x et y, où $\alpha = m(n_x^{-1} + n_y^{-1})^{\frac{1}{2}} + b$.
On calcule d'abord $ss(x,y)$ pour toutes les paires dans un même niveau et on fusionne les noeuds dont le $ss(x,y)$ est le plus petit (et inférieur a α).

Modéle de regroupement local des haplotypes

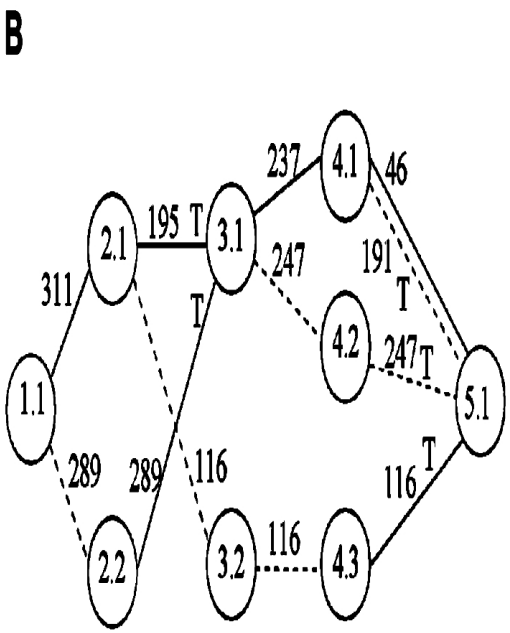
Le modèle local regroupe les haplotypes pour améliorer la prédiction des allèles aux marqueurs $t+1, t+2, \ldots$ sachant les allèles aux marqueurs $t, t-1, \ldots$ dans un haplotype.

Les haplotypes dans un même cluster à la position t vont être généralement dans le même cluster à la position $t+1$, mais pas nécessairement.

Modéle de regroupement local des haplotypes

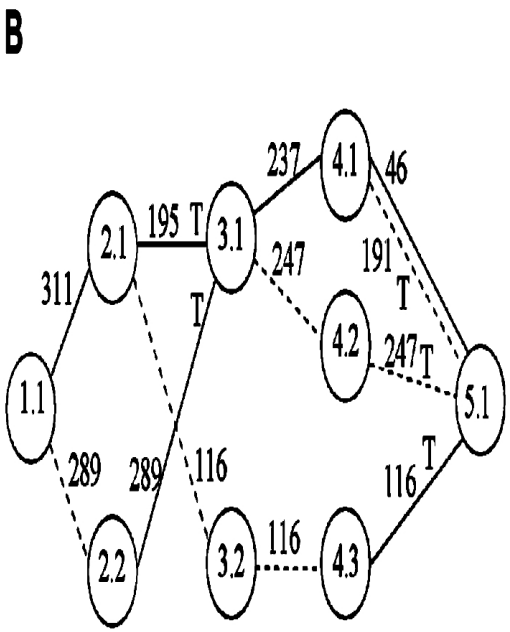
Le modèle local regroupe les haplotypes pour améliorer la prédiction des allèles aux marqueurs $t+1, t+2, \ldots$ sachant les allèles aux marqueurs $t, t-1, \ldots$ dans un haplotype.
Les haplotypes dans un même cluster à la position t vont être généralement dans le même cluster à la position $t+1$, mais pas nécessairement.

Propriétés



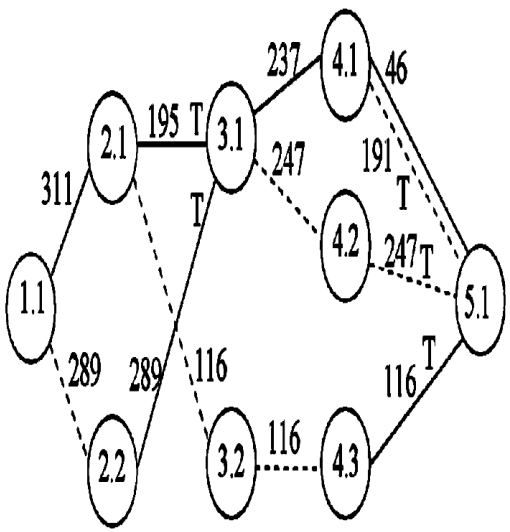
- Le graphe a un noued racine et un noeud final, qui représentent tous les haplotypes
- Le graphe à $D+1$ niveaux. Chaque noeud A a un niveau d. Toutes les arrêtes qui arrivent à A ont un père au niveau $d-1$ et tous ceus qui sortent de A ont des enfants au niveau $d+1$. La racine a pour niveau 0 et le noued final a pour niveau D.

Propriétés

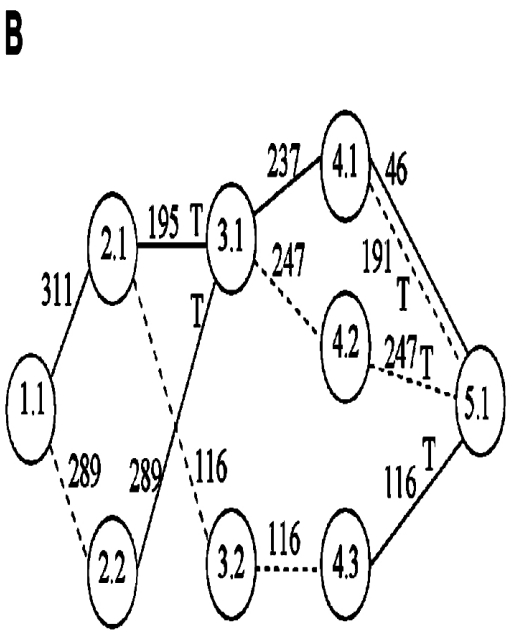


- Le graphe a un noued racine et un noeud final, qui représentent tous les haplotypes
- Le graphe à $D+1$ niveaux. Chaque noeud A a un niveau d. Toutes les arrêtes qui arrivent à A ont un père au niveau $d-1$ et tous ceus qui sortent de A ont des enfants au niveau $d+1$. La racine a pour niveau 0 et le noued final a pour niveau D.

B



- Pour $d = 1, 2, \ldots D$ chaque arrêt avec enfant au niveau d est étiqueté avec un des allèles du marqueur d. Deux arrêtes qui sortent du même père ne peuvent pas être étiquetées avec le même allèle
- Pour chaque haplotype dans l'échantillon, il existe un chemin de la racine jusqu'au noeud final, dont le d-ième allèle est l'étiquette de la d-iéme arrête du chemin. Chaque arrête du graphe a au moins un haplotype qui la traverse.



- Pour $d = 1, 2, \ldots D$ chaque arrêt avec enfant au niveau d est étiqueté avec un des allèles du marqueur d. Deux arrêtes qui sortent du même père ne peuvent pas être étiquetées avec le même allèle
- Pour chaque haplotype dans l'échantillon, il existe un chemin de la racine jusqu'au noeud final, dont le d-ième allèle est l'étiquette de la d-iéme arrête du chemin. Chaque arrête du graphe a au moins un haplotype qui la traverse.

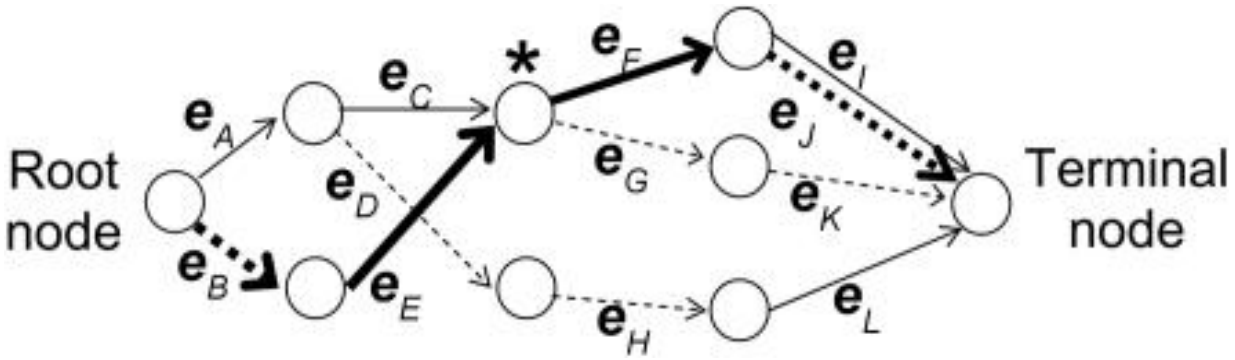


FIGURE: La ligne en gras représente l'haplotype 2112

$n(e)$: nombre d'haplotypes passant par l'arrête e

$n_p(e)$: nombre d'haplotypes passant par le noeud père de e

Exemple :
L'arrête e_F est traversée par les haplotypes 1111, 1112, 2111, 2112.
Le noeud $*$ est le père de e_F. Il est traversé par les mêmes haplotypes que e_F, plus les haplotypes 1122 et 2122.

VLMC, un cas spécial de HMM

But : Voir le modèle comme un HMM
$\Rightarrow$ On peut étendre le modèle aux diplotypes et utiliser les algorithmes d'échantillonnage des HMMs (efficaces).

Le modéle HMM pour haplotypes

état caché : l'arrête

état observé : l'allèle qui étiquette l'arrête

Probabilité d'émission : chaque état émet avec probabilité 1 l'allèle qui étiquette l'arrête. $\Rightarrow$ L'état caché détermine uniquement l'allèle observé, mais l'allèle ne détermine pas l'état caché.

Probabilités de l'état caché initial

$$\begin{aligned} P(e) = & \quad \frac{n(e)}{n_p(e)} \quad \textit{si le père de } e \textit{ est la racine} \\ = & \quad 0 \quad \textit{sinon} \end{aligned}$$

Probabilités de transition

$$\begin{aligned} P(e_1|e_2) = & \quad \frac{n(e_1)}{n_p(e_1)} \quad \textit{si le père de } e_1 \textit{ est l'enfant de } e_2 \\ = & \quad 0 \quad \textit{sinon} \end{aligned}$$

Si les arrêtes e_2 et e_3 ont le même enfant $\Rightarrow$ $P(e_1|e_2) = P(e_1|e_3)$

Probabilités de l'état caché initial

$$\begin{aligned} P(e) = & \quad \frac{n(e)}{n_p(e)} \quad \textit{si le père de } e \textit{ est la racine} \\ = & \quad 0 \quad \textit{sinon} \end{aligned}$$

Probabilités de transition

$$\begin{aligned} P(e_1|e_2) = & \quad \frac{n(e_1)}{n_p(e_1)} \quad \textit{si le père de } e_1 \textit{ est l'enfant de } e_2 \\ = & \quad 0 \quad \textit{sinon} \end{aligned}$$

Si les arrêtes e_2 et e_3 ont le même enfant $\Rightarrow$ $P(e_1|e_2) = P(e_1|e_3)$

HMM Diployde

- On considère des paires d'arrêtes ordonnées à chaque niveau du graphe.
- On peut construire pour chaque état du HMM haployde une classe $L_d,\ d = 1, \ldots D$.
  $L_1 = \{e_A, e_B\},\ L_2 = \{e_C, e_D, e_E\}, \ldots$

Dans le modèle diploïde :

- l'espace des états cachés est $\cup_d$ de $(L_d x L_d)$. Les états cachés sont des paires ordonnées.
- Les états observés sont des paires non ordonnée d'allèles.
- si (e_1, e_2) est l'état caché $\Rightarrow$ le genotype non ordonné déterminé par les allèles qui étiquettent les arrêtes a proba 1.
- en supposant HWE : $\mathbb{P}(e_1, e_2) = \mathbb{P}(e_1)\mathbb{P}(e_2)$ et $\mathbb{P}\left((e_1, e_2)|(e_3, e_4)\right) = \mathbb{P}(e_1|e_3)\mathbb{P}(e_2|e_4)$

HMM Diployde

- On considère des paires d'arrêtes ordonnées à chaque niveau du graphe.
- On peut construire pour chaque état du HMM haployde une classe $L_d,\ d = 1, \ldots D$.
  $L_1 = \{e_A, e_B\},\ L_2 = \{e_C, e_D, e_E\}, \ldots$

Dans le modèle diploïde :

- l'espace des états cachés est $\cup_d$ de $(L_d x L_d)$. Les états cachés sont des paires ordonnées.
- Les états observés sont des paires non ordonnée d'allèles.
- si (e_1, e_2) est l'état caché $\Rightarrow$ le genotype non ordonné déterminé par les allèles qui étiquettent les arrêtes a proba 1.
- en supposant HWE : $\mathbb{P}(e_1, e_2) = \mathbb{P}(e_1)\mathbb{P}(e_2)$ et $\mathbb{P}\left((e_1, e_2)|(e_3, e_4)\right) = \mathbb{P}(e_1|e_3)\mathbb{P}(e_2|e_4)$

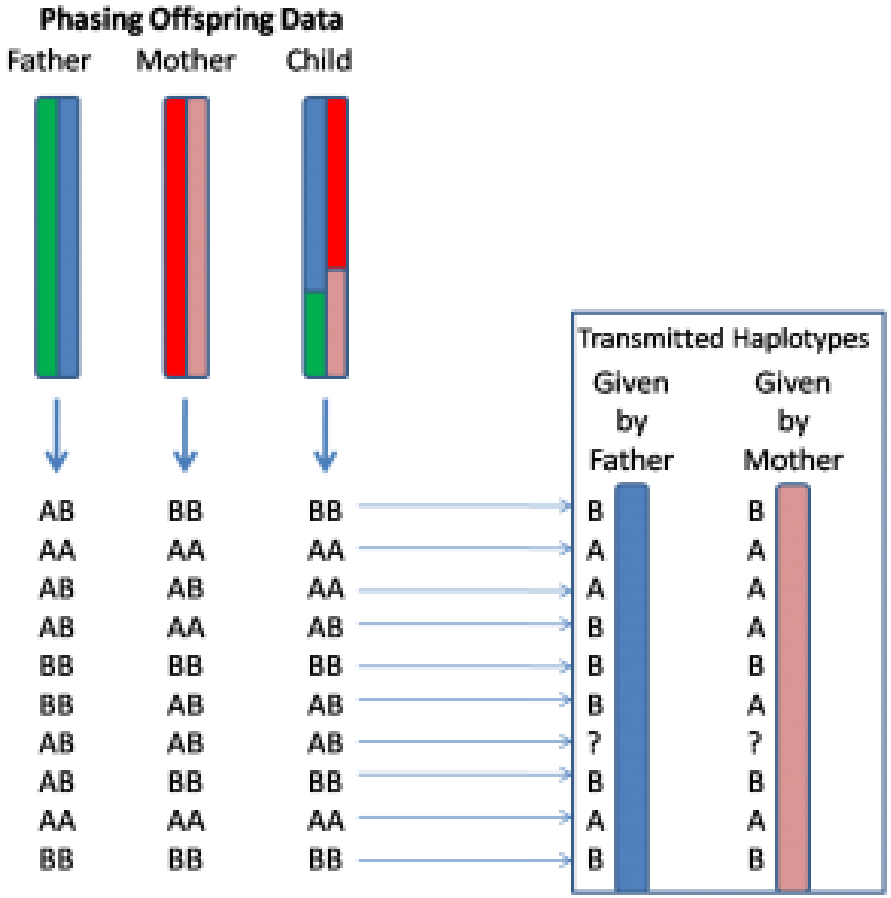
HMM Diployde

- On considère des paires d'arrêtes ordonnées à chaque niveau du graphe.
- On peut construire pour chaque état du HMM haployde une classe $L_d,\ d = 1, \dots D$.
  $L_1 = \{e_A, e_B\},\ L_2 = \{e_C, e_D, e_E\}, \dots$

Dans le modèle diploïde :

- l'espace des états cachés est $\cup_d$ de $(L_d x L_d)$. Les états cachés sont des paires ordonnées.
- Les états observés sont des paires non ordonnée d'allèles.
- si (e_1, e_2) est l'état caché $\Rightarrow$ le genotype non ordonné déterminé par les allèles qui étiquettent les arrêtes a proba 1.
- en supposant HWE : $\mathbb{P}(e_1, e_2) = \mathbb{P}(e_1)\mathbb{P}(e_2)$ et $\mathbb{P}\left((e_1, e_2)|(e_3, e_4)\right) = \mathbb{P}(e_1|e_3)\mathbb{P}(e_2|e_4)$

Algorithme pour phaser les données



L'algorithme pour phaser échantillonne dans le HMM diploïde conditionnellement aux données observés en utilisant un algorithme de type forward-backward

Échantillonner de un HMM

Pour un individu :

- soit g_d le genotype observé non ordonné au marquer m
- soit l'état $s_d = (e_1, e_2)$ une paire ordonnée d'arrêtes dans $L_d x L_d$

$\forall s_d$ dans le HMM diploïde et le genotype de l'individu $\{g_1, g_2, \ldots g_D\}$, on peut définir

Variables Forward

$$\alpha_d(s_d) = \mathbb{P}(g_1, g_2, \ldots, g_d, s_d)$$

On peut calculer α_d par induction

Initiation : $\alpha_1(s_1) = \mathbb{P}(g_1, s_1) = \mathbb{P}(s_1)\mathbb{P}(g_1|s_1)$

Induction :

$$\begin{aligned}
\alpha_{d+1}(s_{d+1}) &= \mathbb{P}(g_1, g_2, \ldots, g_{d+1}, s_{d+1}) \\
&= \sum_{s_d} \mathbb{P}(g_1, g_2, \ldots, g_{d+1}, s_d, s_{d+1}) \\
&= \sum_{s_d} \mathbb{P}(g_1, g_2, \ldots, g_d, s_d)\mathbb{P}(g_{d+1}|s_{d+1})\mathbb{P}(s_{d+1}|s_d) \\
&= \mathbb{P}(g_{d+1}|s_{d+1}) \sum_{s_d} \mathbb{P}(g_1, g_2, \ldots, g_d, s_d)\mathbb{P}(s_{d+1}|s_d)
\end{aligned}$$

Les $\mathbb{P}(g_{d+1}|s_{d+1})$ sont 0 ou 1.

Backward

Échantillonner aux états cachés conditionnellement au genotype de l'individu :

Initiation : Choisir l'état s_D au hasard avec probabilité proportionnelle a $\alpha_D(s_D)$

Induction : sachant les états $s_{d+1}, \ldots, s_D$, choisir l'état s_d avec proba

$$\begin{aligned}
\mathbb{P}(s_d | s_{d+1}, \ldots, s_D, g_1, \ldots, g_D) &= \mathbb{P}(s_d | s_{d+1}, g_1, \ldots, g_{d+1}) \\
&= \mathbb{P}(s_d, s_{d+1}, g_1, \ldots, g_{d+1}) / \alpha_{d+1}(s_{d+1}) \\
&= \mathbb{P}(g_{d+1} | s_{d+1}) \mathbb{P}(s_{d+1} | s_d) \frac{\alpha_d(s_d)}{\alpha_{d+1}(s_{d+1})}
\end{aligned}$$

Le chemin échatillonné des états cachés correspond à une paire ordonnée d'haplotypes qui est compatible avec le genotype de l'individu.

Algorithme pour phaser de Beagle

Début : On phase les individus au hasard.

Chaque itération : on utilise les données phasées pour estimer le DAG.

Après avoir construit le DAG, on échantillonne des haplotypes phasés pour chaque individu dans le modèle HMM diploïde.

On utilise ces données pour commencer une autre itération.

Itération finale : on utilise l'algorithme de Viterbi pour sélectionner l'haplotype le plus probable pour chaque individu.

Algorithme pour phaser de Beagle

Début : On phase les individus au hasard.

Chaque itération : on utilise les données phasées pour estimer le DAG.

Après avoir construit le DAG, on échantillonne des haplotypes phasés pour chaque individu dans le modèle HMM diploïde.

On utilise ces données pour commencer une autre itération.

Itération finale : on utilise l'algorithme de Viterbi pour sélectionner l'haplotype le plus probable pour chaque individu.

Autres applications

- Prédiction de données manquantes.
- Analyses d'association.
- Détection de sélection.