A generalized Viterbi algorithm for identifying the multiple most likely paths of hidden states in a HMM

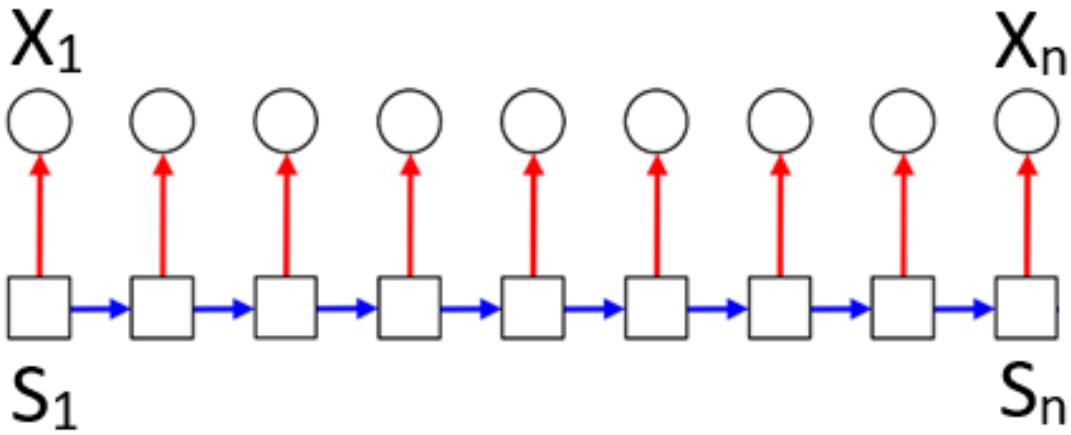
M. Pluntz, S. Foulon, A.-L. Leutenegger, H. Perdry

NeuroDiderot Inserm UMR1141, Université Paris Cité, France, CESP Inserm U1018, Université Paris-Saclay, UVSQ, Villejuif, France.

3 July 2025

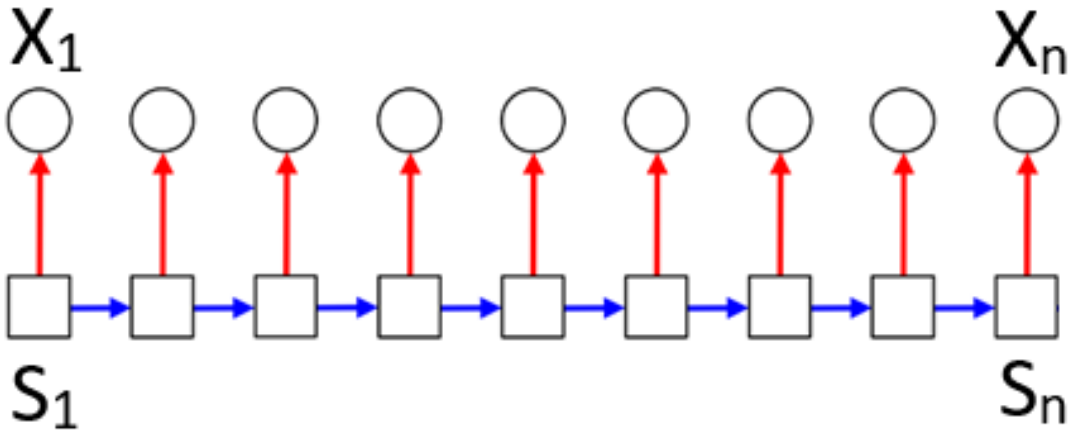
Hidden Markov Model

We observe $(X_k)_k$, which depend on an unobserved Markov chain: the path of hidden states $(S_k)_k$.



Hidden Markov Model

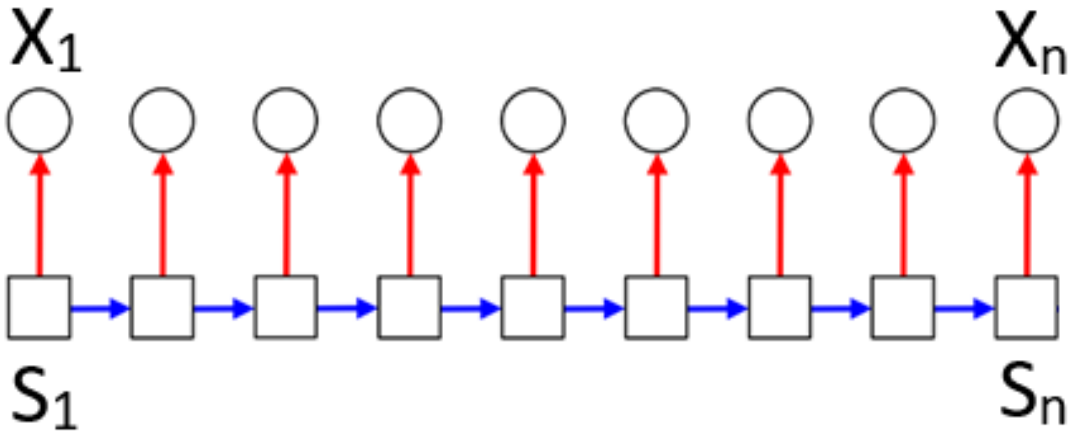
What is the path of hidden states?



- Posterior decoding : path of the most probable hidden states (k by k) given the data.
- Viterbi decoding : most probable global path given the data.

Hidden Markov Model

What is the path of hidden states?

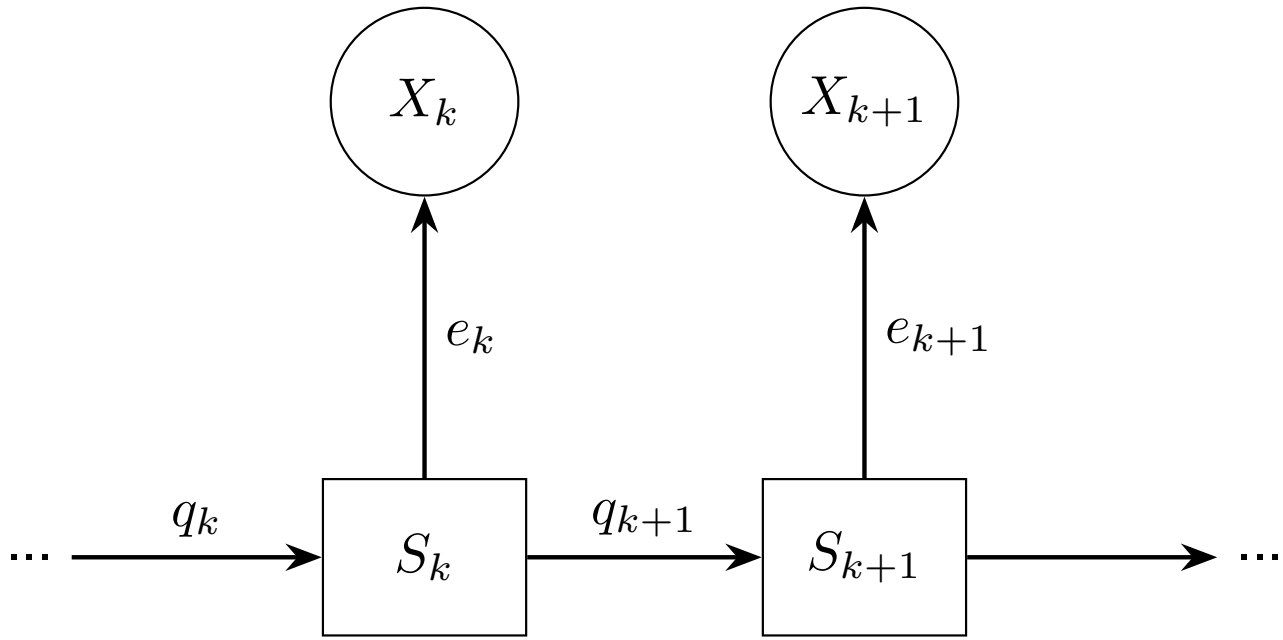


- Posterior decoding : path of the most probable hidden states (k by k) given the data.
- Viterbi decoding : most probable global path given the data.
- $\alpha-$Viterbi decoding : set of most probable global paths (within a ratio α of the best probability) given the data.

Contents

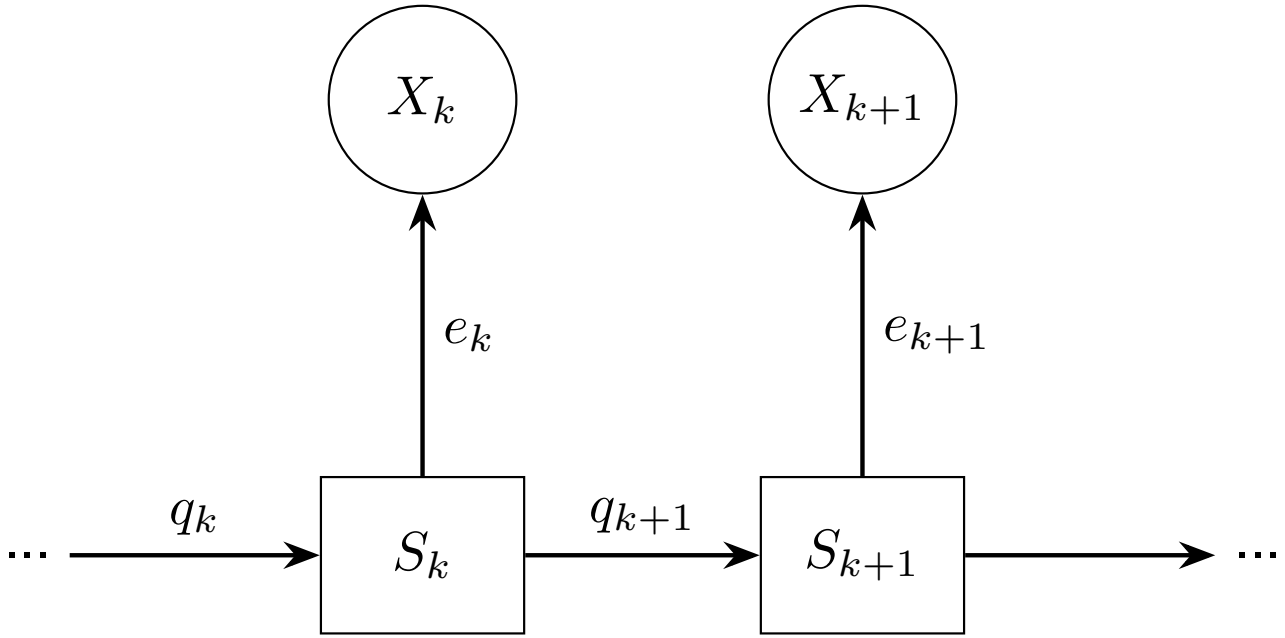
- Details of the HMM
- (simple) Viterbi algorithm
- (multiple) α-Viterbi algorithm
- Application 1: Homozygosity by descent in genetics
- Application 2: Walking the mammoth

Hidden Markov Model



- $k \in \{1, .., n\}$
- Hidden state: $S_k \in \Sigma$. $(S_k)_{1 \leq i \leq n}$ is a Markov chain.
- Initialization: $\forall s \in \Sigma,\ q_1(s) = \mathrm{P}\left(S_1 = s\right)$
- Transition: $q_k(s, s') = \mathrm{P}\left(S_k = s' \mid S_{k-1} = s\right)$
- Emission: $e_k(s, x) = \mathrm{P}\left(X_k = x \mid S_k = s\right)$

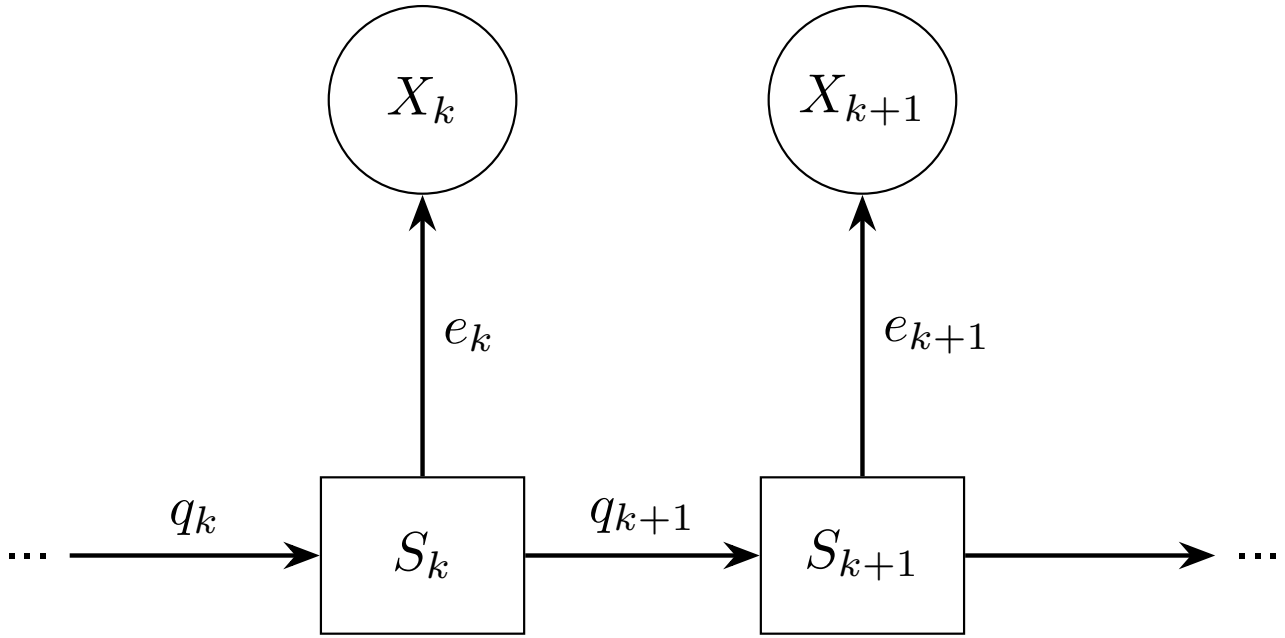
Hidden Markov Model



Joint distribution of (S, X):

$$\mathrm{P}\,(S, X) = q_1(S_1)e_1(S_1, X_1) \prod_{k=2}^{n} q_k(S_{k-1}, S_k)e_1(S_k, X_k)$$

Hidden Markov Model



Joint distribution of (S, X):

$$\mathrm{P}\,(S, X) = q_1(S_1)e_1(S_1, X_1)\prod_{k=2}^{n} q_k(S_{k-1}, S_k)e_1(S_k, X_k)$$

All the probabilities are known. If the model is parametric, we assume the parameters are already estimated by a suitable method: Expectation-maximization (Baum-Welch), Direct likelihood maxmization, etc.

Viterbi algorithm

$$\mathrm{P}\left(S, X\right) = q_1(S_1)e_1(S_1, X_1)\prod_{k=2}^{n} q_k(S_{k-1}, S_k)e_1(S_k, X_k)$$

Observing $X = x$, we look for the $\sigma \in \Sigma^n$ which maximizes $\mathrm{P}\left(S = \sigma, X = x\right)$.

Viterbi algorithm - forward probabilities

$$\mathrm{P}\left(S, X\right) = q_1(S_1)e_1(S_1, X_1)\prod_{k=2}^{n} q_k(S_{k-1}, S_k)e_1(S_k, X_k)$$

Observing $X = x$, we look for the $\sigma \in \Sigma^n$ which maximizes $\mathrm{P}\left(S = \sigma, X = x\right) = p^{(n)}(\sigma)$ where:

$\forall k \in \{1, .., n\}, \forall \sigma \in \Sigma^k,$

$$\begin{aligned} p^{(k)}(\sigma) &:= \mathrm{P}\left(S_{\{1,..,k\}} = \sigma, X_{\{1,..,k\}} = x_{\{1,..,k\}}\right) \\ &= q_1(\sigma_1)e_1(\sigma_1, x_1)\prod_{i=2}^{k} q_i(\sigma_{i-1}, \sigma_i)e_1(\sigma_i, x_i) \end{aligned}$$

Maximal probability: $p_M = \max_{\sigma \in \Sigma^n} p^{(n)}(\sigma)$.

Viterbi algorithm - quantities of interest

- Maximal forward probability reaching a given state at a given step:

$$\forall k \in \{1, .., n\}, \forall s \in \Sigma,$$

$$v(k, s) := \max \left\{ p^{(k)}(\sigma) : \sigma \in \Sigma^k, \sigma_k = s \right\}$$

Viterbi algorithm - quantities of interest

- Maximal forward probability reaching a given state at a given step:

$$\forall k \in \{1, .., n\}, \forall s \in \Sigma,$$

$$v(k, s) := \max \left\{ p^{(k)}(\sigma) : \sigma \in \Sigma^k, \sigma_k = s \right\}$$

- Traceback: predecessor reaching the maximal forward probability

$$\forall k \in \{2, .., n\}, \forall s \in \Sigma,$$

$\tau(k, s) :\in \Sigma$ such that : $\exists\, \sigma \in \Sigma^k, \sigma_k = s, \sigma_{k-1} = \tau(k, s),\ p^{(k)}(\sigma) = v(k, s).$

Viterbi algorithm - quantities of interest

- Maximal forward probability reaching a given state at a given step:

$$\forall k \in \{1, .., n\}, \forall s \in \Sigma,$$

$$v(k, s) := \max \left\{ p^{(k)}(\sigma) : \sigma \in \Sigma^k, \sigma_k = s \right\}$$

- Traceback: predecessor reaching the maximal forward probability

$$\forall k \in \{2, .., n\}, \forall s \in \Sigma,$$

$\tau(k, s) :\in \Sigma$ such that : $\exists\, \sigma \in \Sigma^k, \sigma_k = s, \sigma_{k-1} = \tau(k, s),\ p^{(k)}(\sigma) = v(k, s).$

- Optimal path tail:

$$\forall k \in \{1, .., n\},$$

$\sigma^{(k:n)} :\in \Sigma^{n-k+1}$ such that : $\exists \sigma \in \Sigma^{k-1}, p^{(n)}(\sigma \cdot \sigma^{(k:n)}) = p_M.$

Viterbi algorithm - computing the quantities of interest

$$v(k,s) := \max\left\{p^{(k)}(\sigma) : \sigma \in \Sigma^k, \sigma_k = s\right\}$$

$$\exists\, \sigma \in \Sigma^k, \sigma_k = s, \sigma_{k-1} = \tau(k,s),\ p^{(k)}(\sigma) = v(k,s).$$

- Forward recurrence for the maximal forward probabilities and tracebacks:

$$v(1,s) = q_1(s)e_1(s,x_1)$$

$$v(k+1,s) = \max_{t\in\Sigma}\left\{v(k,t)\ q_{k+1}(t,s)\right\}\ e_{k+1}(s,x_{k+1})$$

$$\tau(k+1,s) \in \operatorname*{argmax}_{t\in\Sigma}\left\{v(k,t)\ q_{k+1}(t,s)\right\}$$

Viterbi algorithm - computing the quantities of interest

- Forward recurrence for the maximal forward probabilities and tracebacks:

$$v(1, s) = q_1(s) e_1(s, x_1)$$

$$v(k+1, s) = \max_{t \in \Sigma} \{v(k, t)\ q_{k+1}(t, s)\}\ \ e_{k+1}(s, x_{k+1})$$

$$\tau(k+1, s) \in \operatorname*{argmax}_{t \in \Sigma} \{v(k, t)\ q_{k+1}(t, s)\}$$

- Backward recurrence for the optimal path, using the final vector $v(n, \cdot)$ and the tracebacks:

$$\sigma^{(n:n)} :\in \operatorname*{argmax}_{s \in \Sigma} v(n, s)$$

$$\sigma^{((k-1):n)} := \tau(k, \sigma_1^{(k:n)}) \cdot \sigma^{(k:n)}$$

We reach an optimal full path $\sigma^{(1:n)}$.

Viterbi algorithm

- Fast: $O\left(n \times |S|^2\right)$ complexity
- Not very memory-intensive: the largest object is the $n \times |S|$ matrix τ. The $v(k, \cdot)$ can overwrite each other.
- Produces only one path, which is limited for inference and visualization/exploration.

Multiple Viterbi algorithm (α-Viterbi)

Let $\alpha \in]0, 1]$. We look for all the paths $\sigma \in \Sigma^n$ such that $p^{(n)}(\sigma) \geq \alpha\, p_M$.

Variant: k most probable paths with k fixed in advance. Brown and Golod, Decoding HMMs using the k best paths: algorithms and applications, BMC Bioinformatics, 2010.

α-Viterbi: quantities of interest

- Probability ratio associated with a given transition

$$\forall k \geq 2, \forall (t,q) \in \Sigma^2 \text{ with } v(k,s) > 0 :$$

$$r(k,t,s) := \frac{1}{v(k,s)} \max \left\{ p^{(k)}(\sigma), \sigma \in \Sigma^k : \sigma_{k-1} = t, \sigma_k = s \right\}$$

By construction, $0 \leq r(k,t,s) \leq 1$ and $r(k, \tau(k,s), s) = 1$.

$r(k,t,s)$ indicates how much likelihood is lost by each transition $t \to s$ reaching s at step k, compared to the most likely transition $\tau(k,s) \to s$.

α-Viterbi: quantities of interest

- Probability ratio associated with a given transition

$$\forall k \geq 2, \forall (t,q) \in \Sigma^2 \text{ with } v(k,s) > 0 :$$

$$r(k,t,s) := \frac{1}{v(k,s)} \max \left\{ p^{(k)}(\sigma), \sigma \in \Sigma^k : \sigma_{k-1} = t, \sigma_k = s \right\}$$

- Maximal relative probability of a path tail

$$\forall \sigma^{\text{tail}} \in \Sigma^{n-k+1}, r_M^{(k:n)}(\sigma^{\text{tail}}) := \frac{1}{p_M} \max \left\{ p^{(n)}(\sigma^{\text{head}} \cdot \sigma^{\text{tail}}), \sigma^{\text{head}} \in \Sigma^{k-1} \right\}.$$

By construction, $0 \leq r_M^{(k:n)}(\sigma^{\text{tail}}) \leq 1$ and $r_M^{(k:n)}(\sigma^{(k:n)}) = 1$.

$r_M^{(k:n)}$ indicates how much likelihood is lost by choosing a specific path tail compared to the optimal path.

α-Viterbi: quantities of interest

- Probability ratio associated with a given transition

$$\forall k \geq 2, \forall (t,q) \in \Sigma^2 \text{ with } v(k,s) > 0 :$$

$$r(k,t,s) := \frac{1}{v(k,s)} \max \left\{ p^{(k)}(\sigma), \sigma \in \Sigma^k : \sigma_{k-1} = t, \sigma_k = s \right\}$$

- Maximal relative probability of a path tail

$$\forall \sigma^{\text{tail}} \in \Sigma^{n-k+1}, r_M^{(k:n)}(\sigma^{\text{tail}}) := \frac{1}{p_M} \max \left\{ p^{(n)}(\sigma^{\text{head}} \cdot \sigma^{\text{tail}}), \sigma^{\text{head}} \in \Sigma^{k-1} \right\}.$$

- Set of α-admissible path tails

$$\begin{aligned} E_\alpha^{(k:n)} &:= \left\{ \sigma^{\text{tail}} \in \Sigma^{n-k+1} : \exists\, \sigma^{\text{head}} \in \Sigma^{k-1}, p^{(n)}(\sigma^{\text{head}} \cdot \sigma^{\text{tail}}) \geq \alpha\, p_M \right\} \\ &= \left\{ \sigma^{\text{tail}} \in \Sigma^{n-k+1} : r_M^{(k:n)}(\sigma^{\text{tail}}) \geq \alpha \right\}. \end{aligned}$$

We are looking for $E_\alpha = E_\alpha^{(1:n)}$.

α-Viterbi - computing the quantities of interest

- Forward recurrence for v (like in simple Viterbi) and r:

$$v(1, s) = q_1(s)e_1(s, x_1)$$

$$v(k+1, s) = \max_{t \in \Sigma} \{v(k, t)\ q_{k+1}(t, s)\}\ \ e_{k+1}(s, x_{k+1})$$

$$r(k+1, t, s) = \frac{v(k, t)\ q_{k+1}(t, s)\ e_{k+1}(s)}{v(k+1, s)}.$$

α-Viterbi - computing the quantities of interest

- Forward recurrence for v and r.
- Backward recurrence for the sets of admissible path tails and their maximal relative probabilities, using the final vector $v(n, \cdot)$ and the probability ratios:

$$E_{\alpha}^{(n:n)} := \{(s) : s \in \Sigma, v(n, s) \geq \alpha \, p_M\}$$

$$\forall (s) \in E_{\alpha}^{(n:n)}, \; r_M^{(n:n)}((s)) = \frac{v(n, s)}{p_M}$$

α-Viterbi - computing the quantities of interest

- Forward recurrence for v and r.
- Backward recurrence for the sets of admissible path tails and their maximal relative probabilities, using the final vector $v(n, \cdot)$ and the probability ratios:

$$E_\alpha^{(n:n)} := \{(s) : s \in \Sigma, v(n, s) \geq \alpha\, p_M\}$$

$$\forall (s) \in E_\alpha^{(n:n)},\ r_M^{(n:n)}\left((s)\right) = \frac{v(n, s)}{p_M}$$

$$E_\alpha^{((k-1):n)} = \left\{t \cdot \sigma, t \in \Sigma, \sigma \in E_\alpha^{(k:n)} : r(k, t, \sigma_1)\, r_M^{(k:n)}(\sigma) \geq \alpha\right\}$$

$$\forall \sigma \in E_\alpha^{((k-1):n)},\ r_M^{((k-1):n)}(\sigma) = r(k, \sigma_1, \sigma_2)\, r_M^{(k:n)}(\sigma_{2,..}).$$

Until reaching $E_\alpha = E_\alpha^{(1:n)}$.

α-Viterbi - computing the quantities of interest

- Forward recurrence for v and r.
- Backward recurrence for the sets of admissible path tails:

$$
\begin{aligned}
E_\alpha^{((k-1):n)} &= \left\{ t \cdot \sigma, t \in \Sigma, \sigma \in E_\alpha^{(k:n)} : r(k,t,\sigma_1)\, r_M^{(k:n)}(\sigma) \geq \alpha \right\} \\
&= \bigcup_{\sigma \in E_\alpha^{(k:n)}} \left\{ t \cdot \sigma, t \in \Sigma : r(k,t,\sigma_1) \;\geq \frac{\alpha}{r_M^{(k:n)}(\sigma)} \right\}.
\end{aligned}
$$

$E_\alpha^{(k:n)}$ grows in size as k decreases, with every tail having at least one antecedent.

If a bound is desired on $|E_\alpha|$, we keep $|E_\alpha^{(k:n)}|$ under it by increasing α when necessary.

Cumulative probabilities

Given α and E_α, we can compute the total probability of the set:

$$p^{(n)}\left(E_\alpha\right) = \sum_{\sigma \in E_\alpha} p^{(n)}(\sigma)$$

How does it behave as a function of α?

How to choose α to get a desired $p^{(n)}\left(E_\alpha\right)$?

Cumulative probabilities

$$v(k,s) = \max\left\{p^{(k)}(\sigma) : \sigma \in \Sigma^k, \sigma_k = s\right\}$$

$$D_\alpha^{(1:k)}(s) := \left\{\sigma \in \Sigma^k : \sigma_k = s,\ p^{(k)}(\sigma) \geq \alpha\, v(k,s)\right\}$$

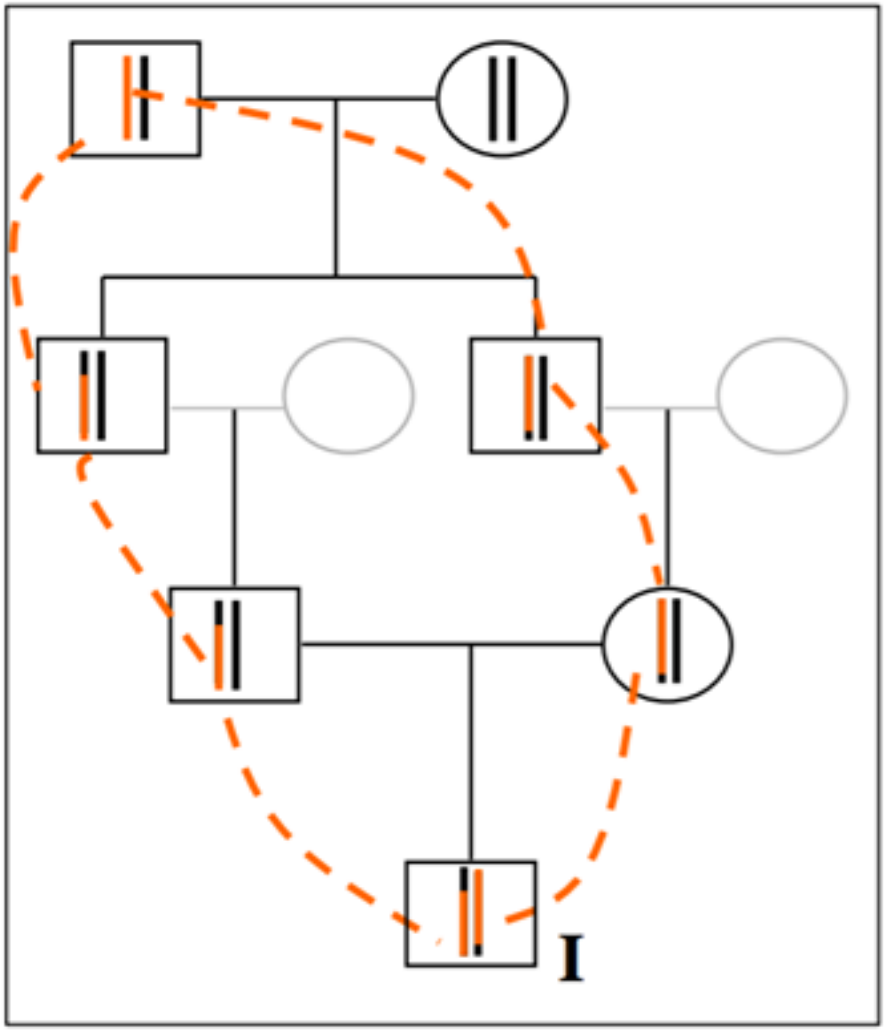
$$\mathrm{CP}(k,s,\alpha) := \frac{1}{v(k,s)} \sum_{\sigma \in D_\alpha^{(1:k)}(s)} p^{(k)}(\sigma)$$

The $\mathrm{CP}(k,s,\alpha)$ (on a logarithmic grid of α) are computed by forward recurrence without having to enumerate the $D_\alpha^{(1:k)}(s)$:

$$\mathrm{CP}(k+1,s,\alpha) = \sum_{t \in \Sigma} \mathrm{CP}\left(k, t, \frac{\alpha}{r(k+1,t,s)}\right) r(k+1,t,s)$$

Finally, we get $p^{(n)}\left(E_\alpha\right)$ by summing over the $\mathrm{CP}(n,s,\alpha)$.

Homozygosity by descent



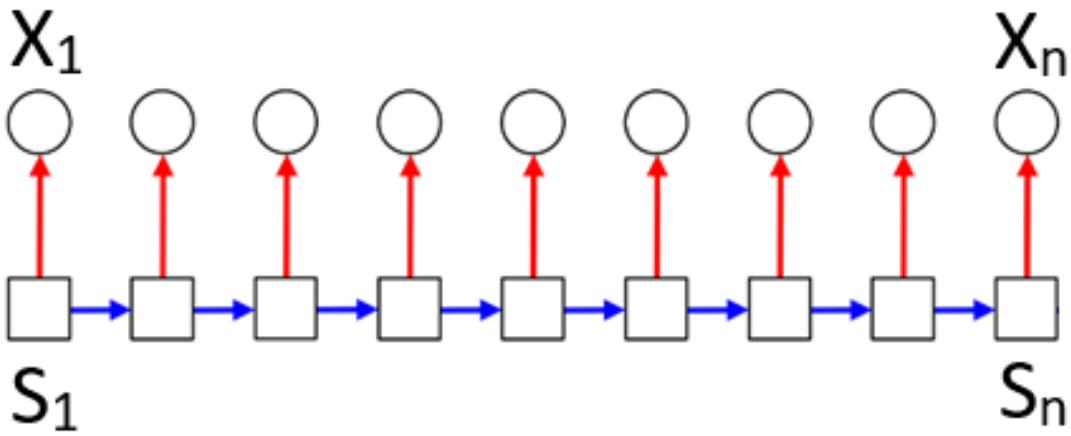
Child of first cousins.

Homozygosity: Identical alleles at a locus of the genome.

HBD: child of related parents may inherit an ancestor's allele through both parents.

At each generation, the two chromosomes of a pair randomly exchange segments (crossing-over). Therefore, HBD happens in randomly started or interrupted segments.

Markov Model for HBD



$k = 1, .., n$: loci

$S_k \in \{\text{nh}, \text{h}\}$: non-HBD or HBD

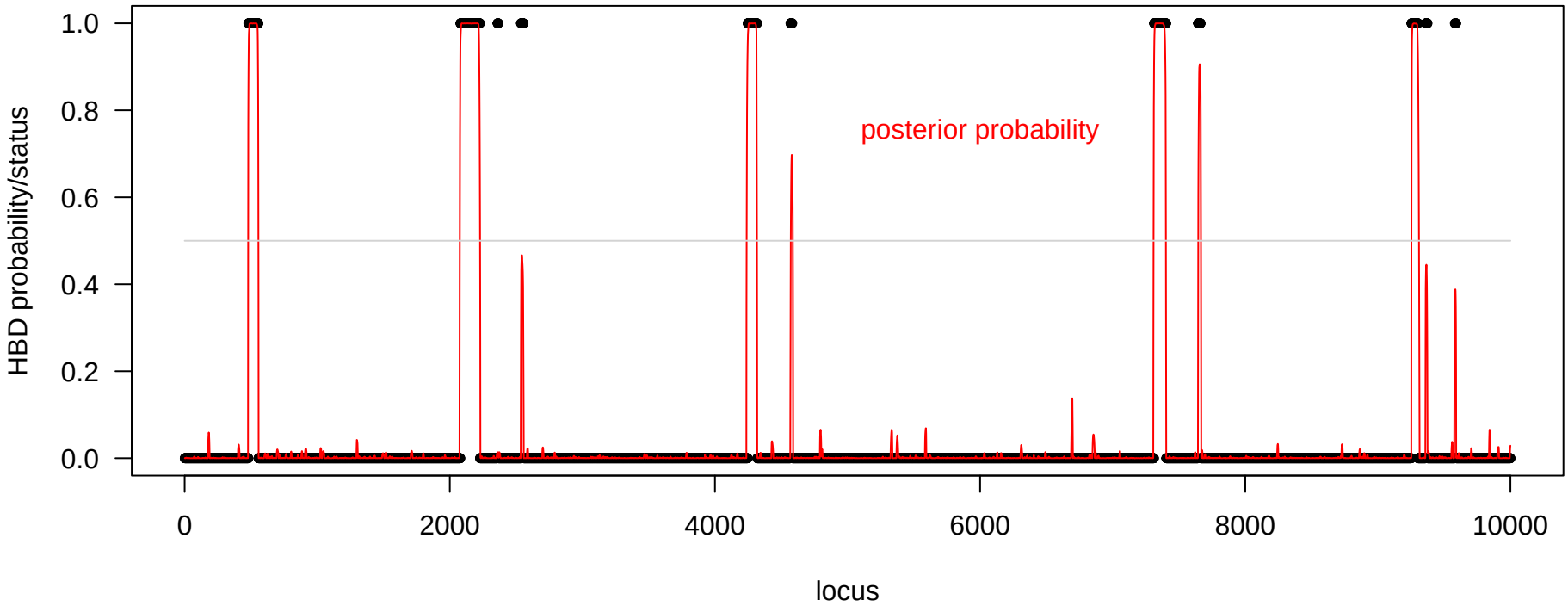
$r_{\text{nh}\to\text{h}}$, $r_{\text{h}\to\text{nh}}$:
rates of state switching per Morgan (genetic distance unit)

First cousins: $r_{\text{nh}\to\text{h}} = 2/5, \quad r_{\text{h}\to\text{nh}} = 6$

Observed alternative alleles :

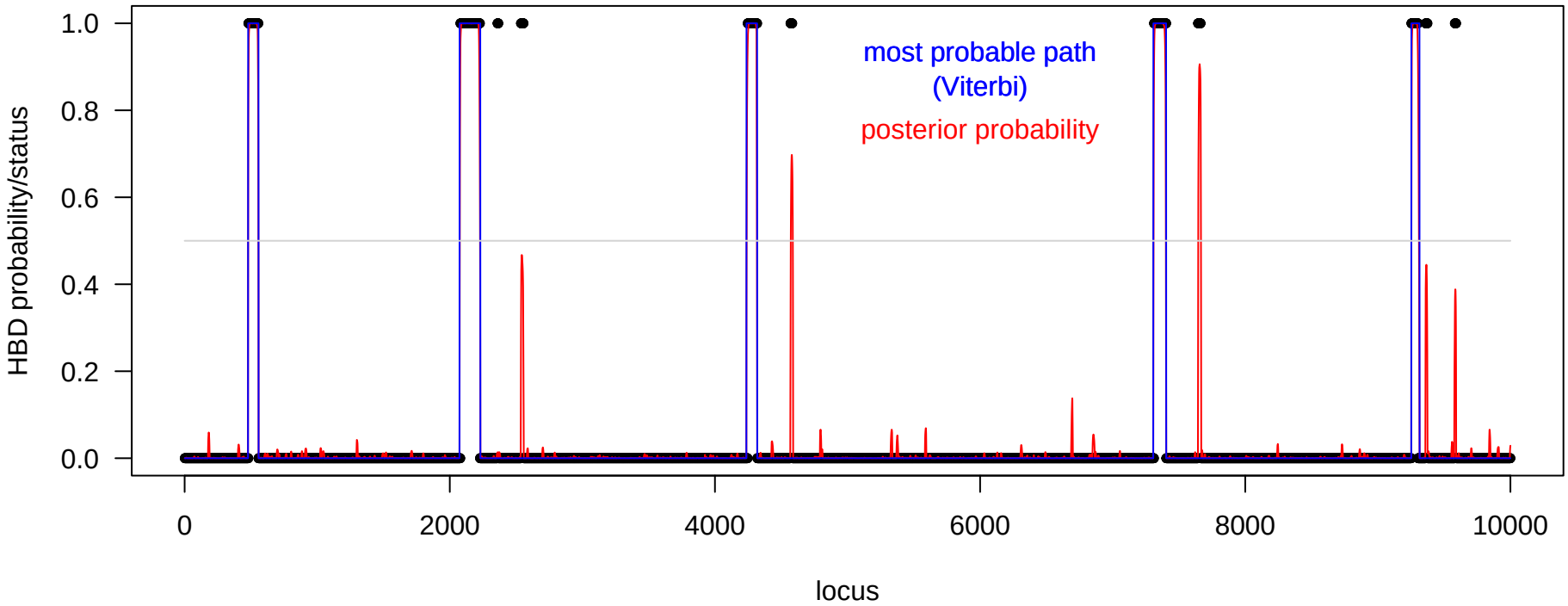
$X_k \in \{0, 1, 2\}$ with emission probabilities dependent on S_k, allele frequencies, and genotyping error rate.

Markov Model for HBD - comparision of decodings



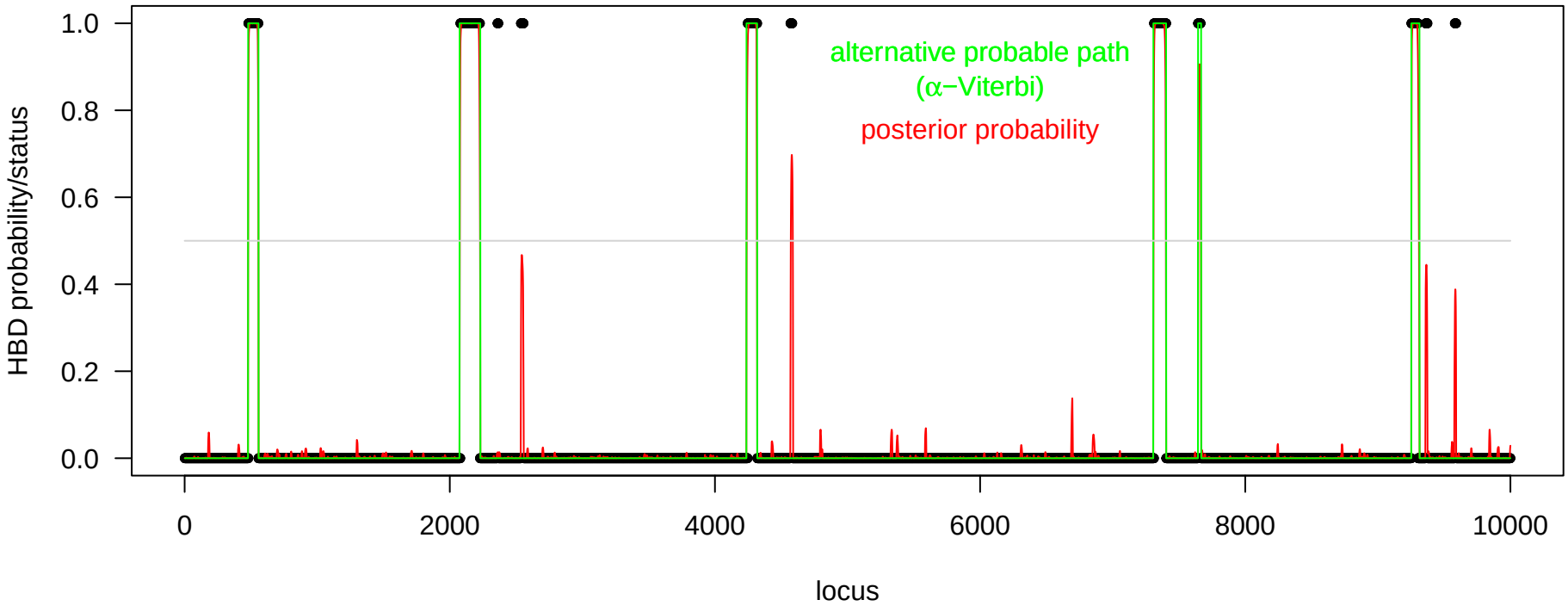
HBD status of a simulated child of first cousins

Markov Model for HBD - comparision of decodings



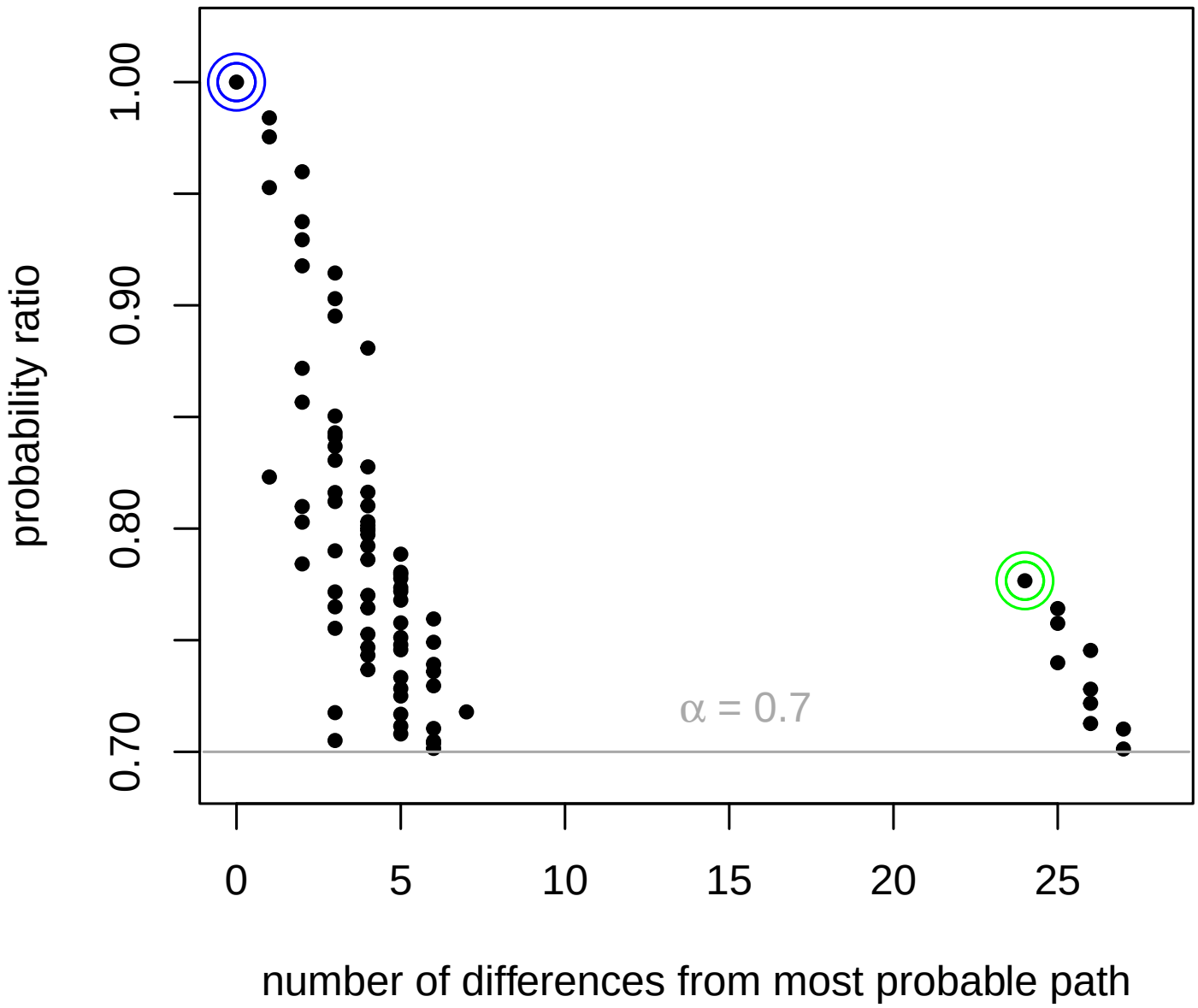
HBD status of a simulated child of first cousins

Markov Model for HBD - comparision of decodings

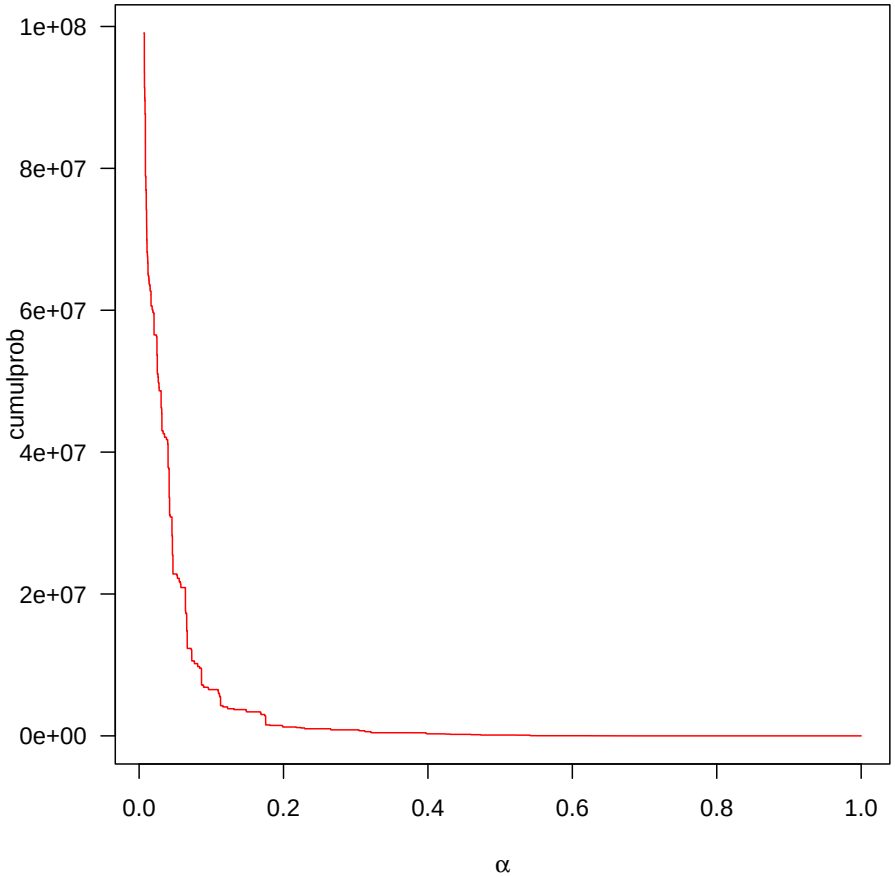


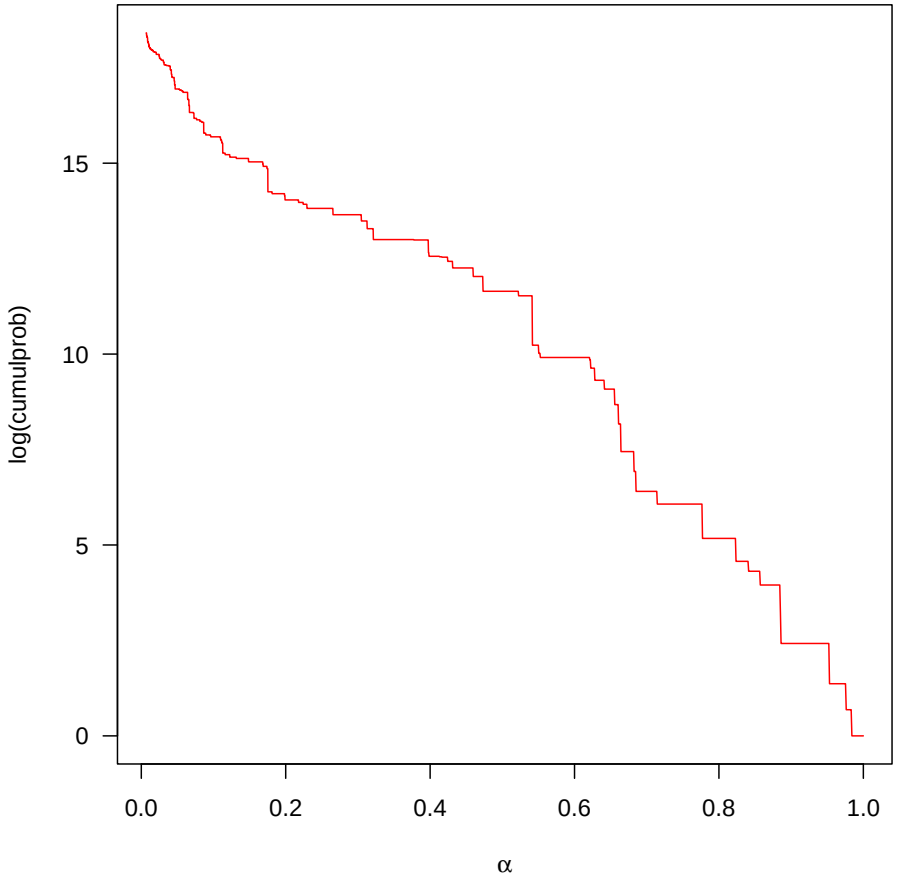
HBD status of a simulated child of first cousins

Markov Model for HBD - comparision of decodings

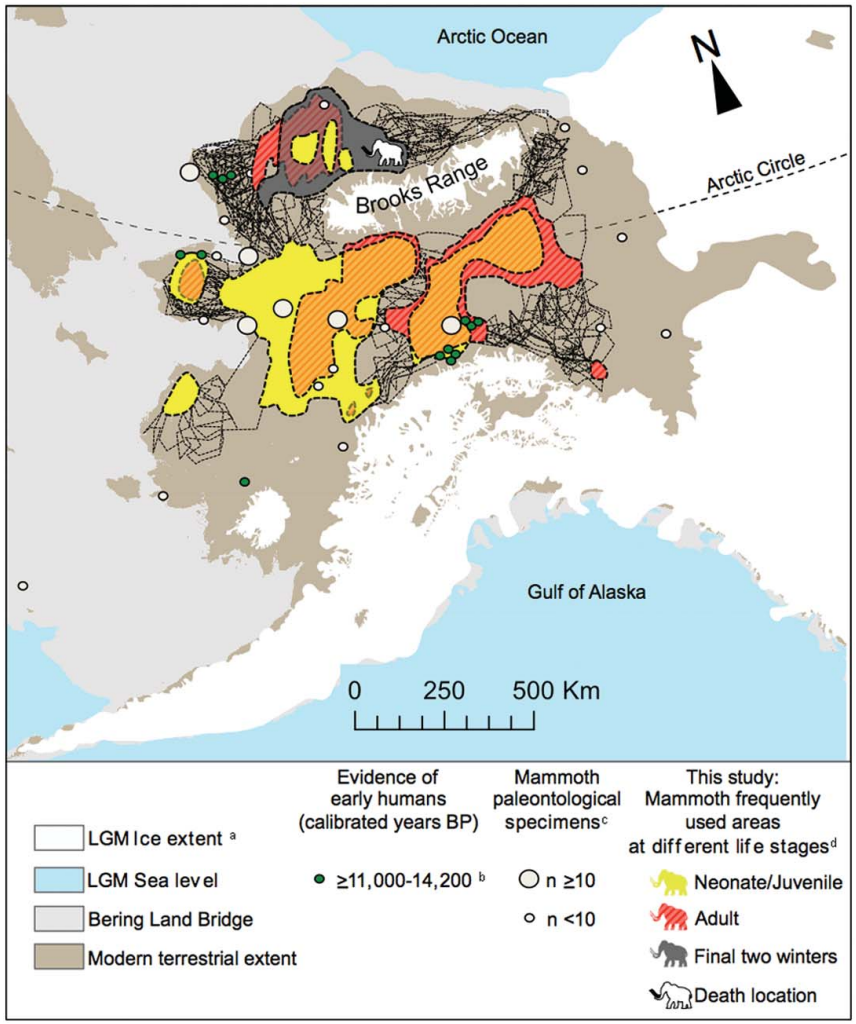


Markov Model for HBD - cumulative probabilities





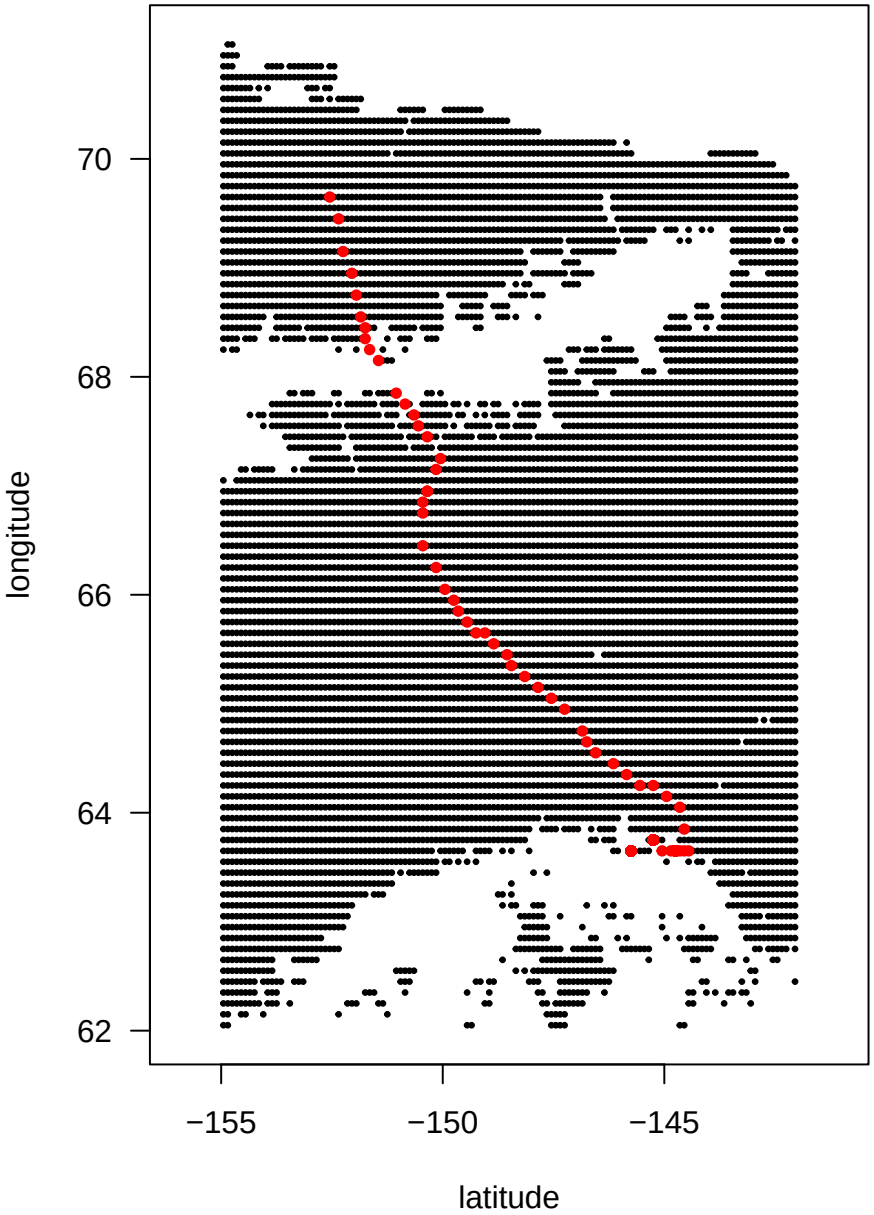
Walking the mammoth

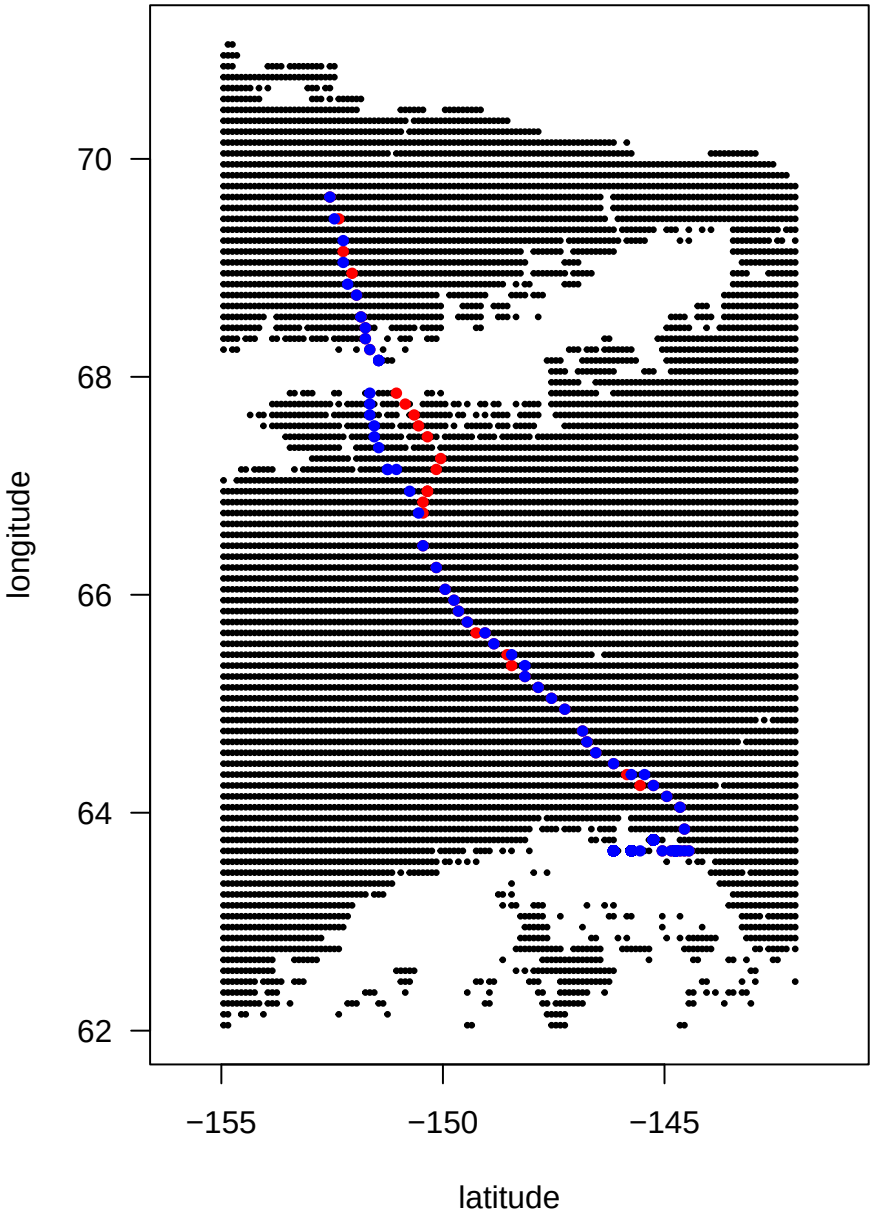


A mammoth's body was found with isotope concentrations in his tusk mirroring those of his environment accross his life. Where was he?

Wooller et al., Lifetime mobility of an Arctic woolly mammoth, Science, 2021.

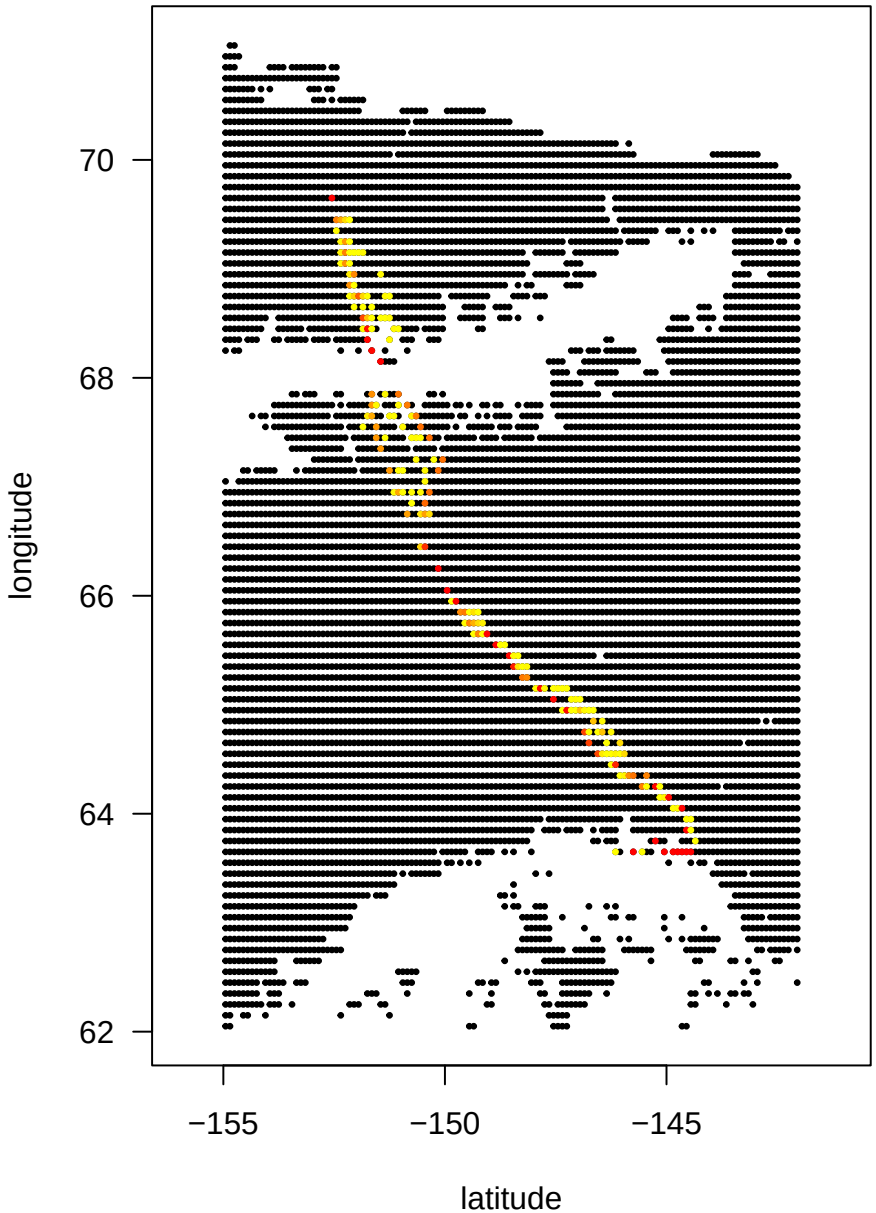
Walking the mammoth





Walking the mammoth

All trajectories in $E_{1/2}^{(1:n)}$. Colors indicate density of paths.



Conclusion

- We propose a Viterbi-like algorithm for finding all the paths of hidden states with posterior probability above a given proportion of the maximal posterior probability (that of the Viterbi path).

- The cumulative probability function allows to choose the cutoff α to target.
- It also shows that even large families of paths may gather only a little cumulative probability.
- We used α-Viterbi on both a small-Σ (HBD) and large Σ (Mammoth) problem, with differing computational challenges.
- Data storage question: how to represent a large family of long similar paths?
- Comparision with Brown and Golod's k best paths?

Thank you