
Detecting recombination in DNA sequence alignments

Dirk Husmeier

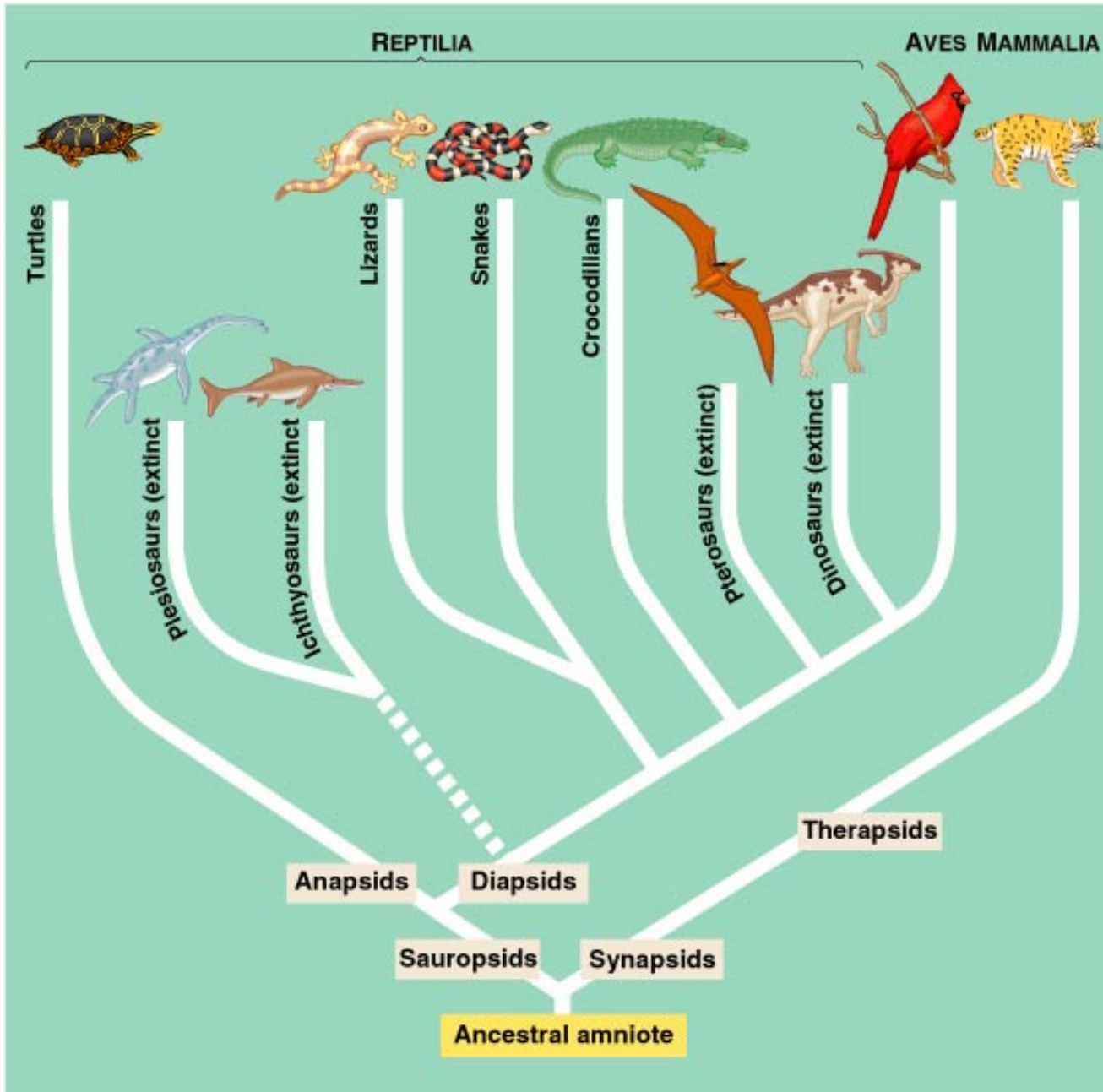
Biomathematics and Statistics Scotland

Edinburgh, United Kingdom

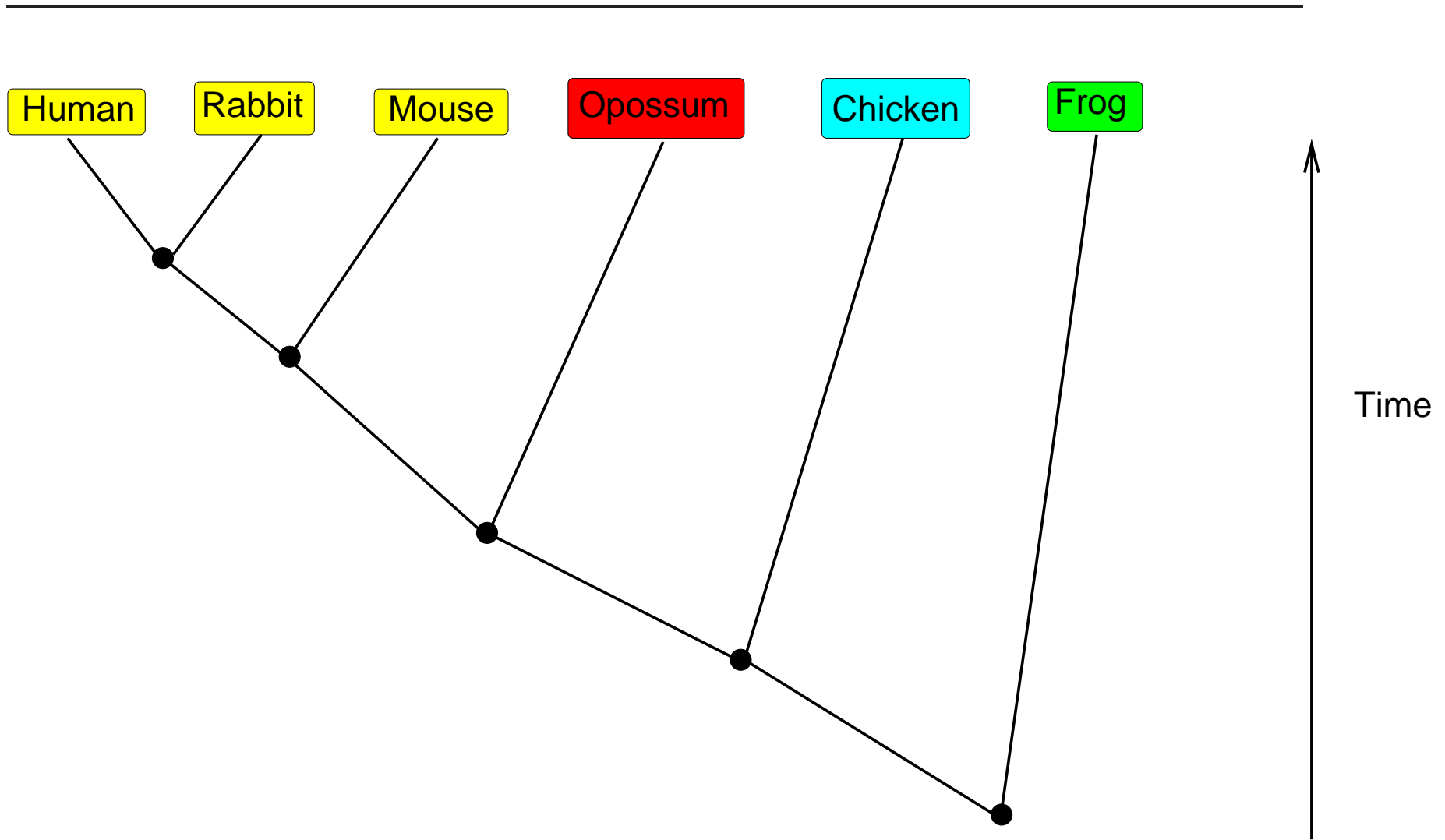
Email: dirk@bioss.ac.uk

<http://www.bioss.ac.uk/~dirk>

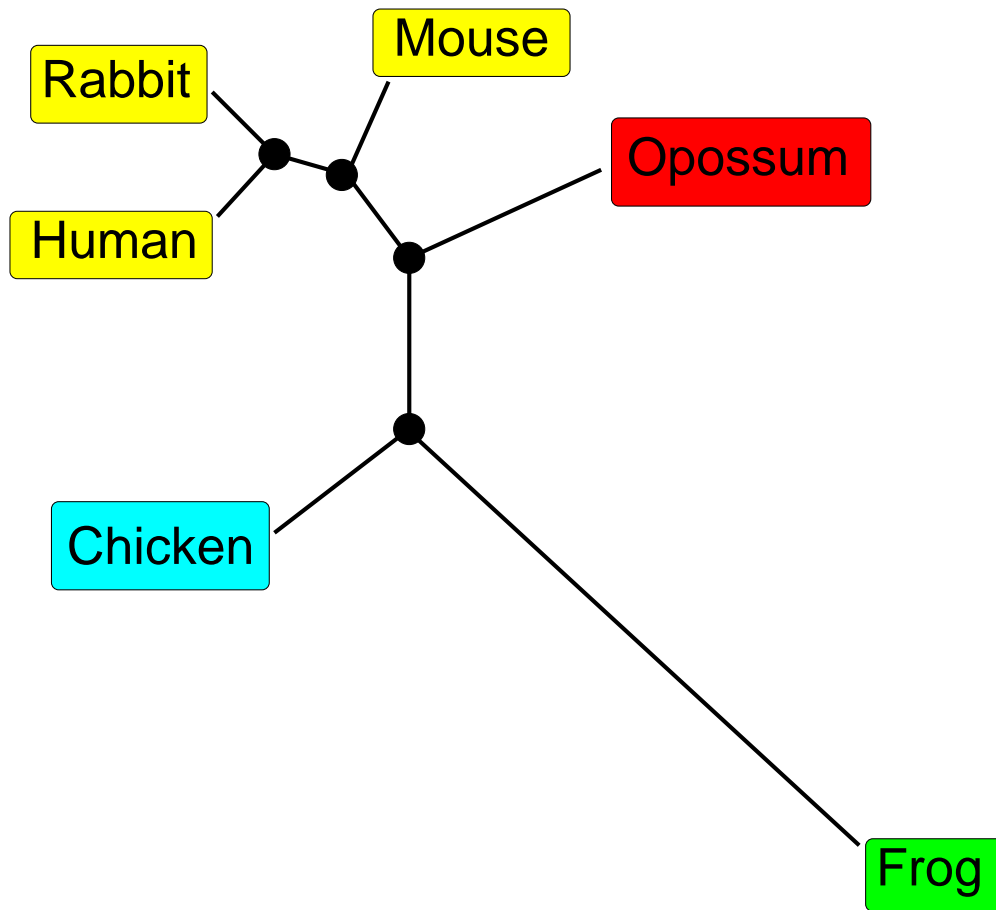
- Recapitulation: Molecular evolution and phylogeny
- Detection of recombination



Rooted Phylogenetic Tree



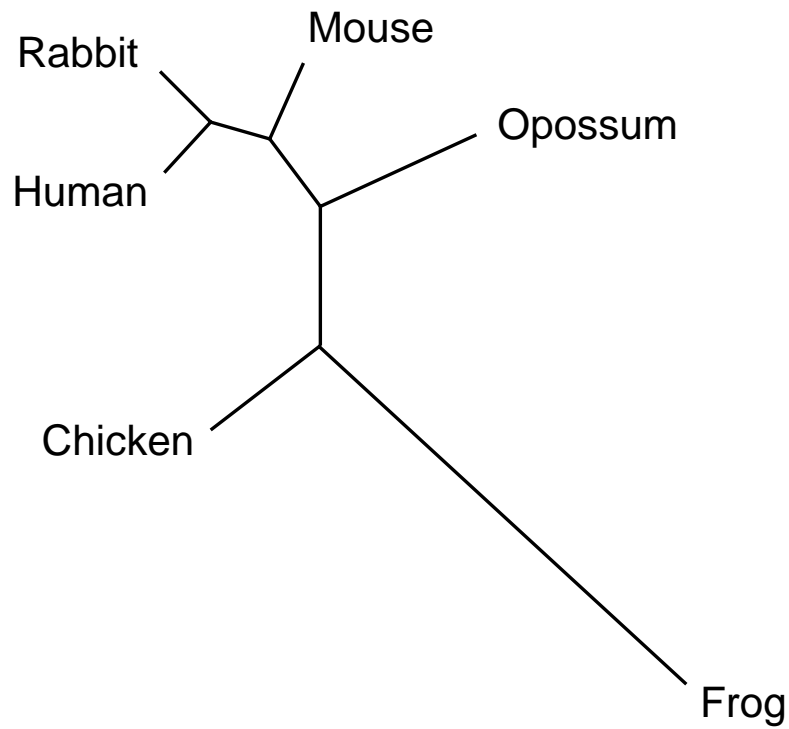
Unrooted Phylogenetic Tree



--> Topology

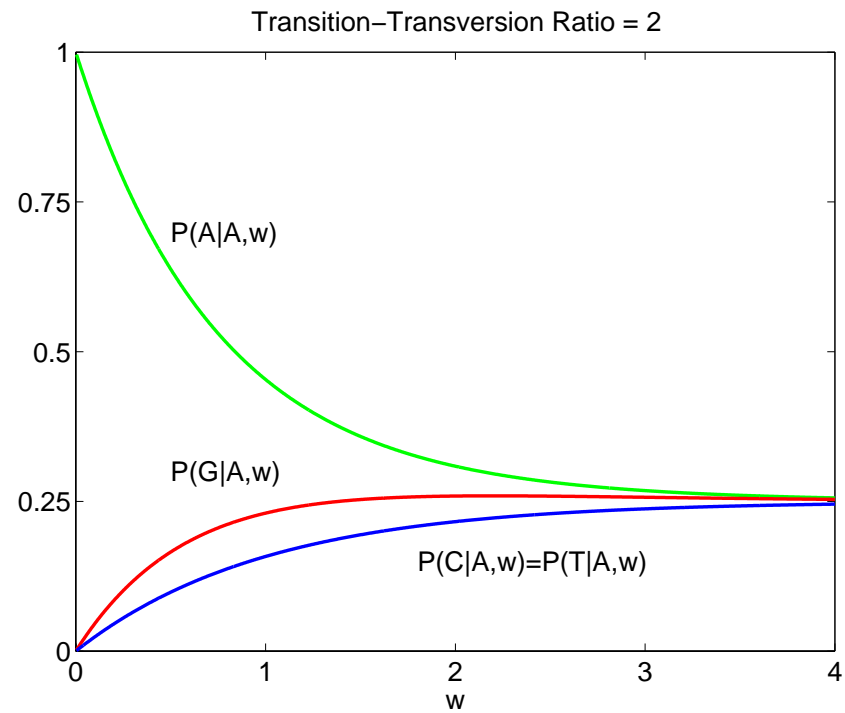
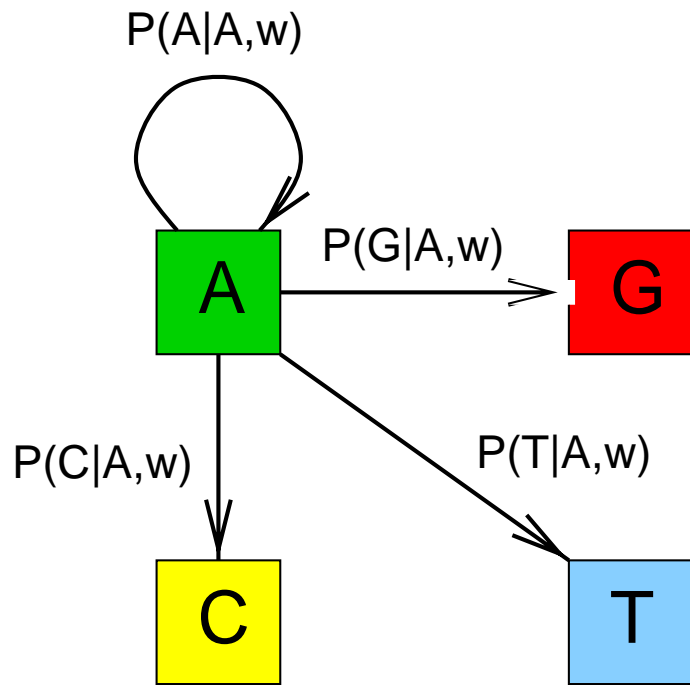
--> Branch lengths

Frog	G	C	T	T	G	A	C	T	T	C	T	G	A	G	G	T	T
Chicken	G	C	G	T	A	A	C	T	T	C	A	C	A	T	G	A	T
Human	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T
Rabbit	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T
Mouse	G	C	G	T	C	A	C	T	T	G	A	C	A	G	G	C	T
Opossum	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T

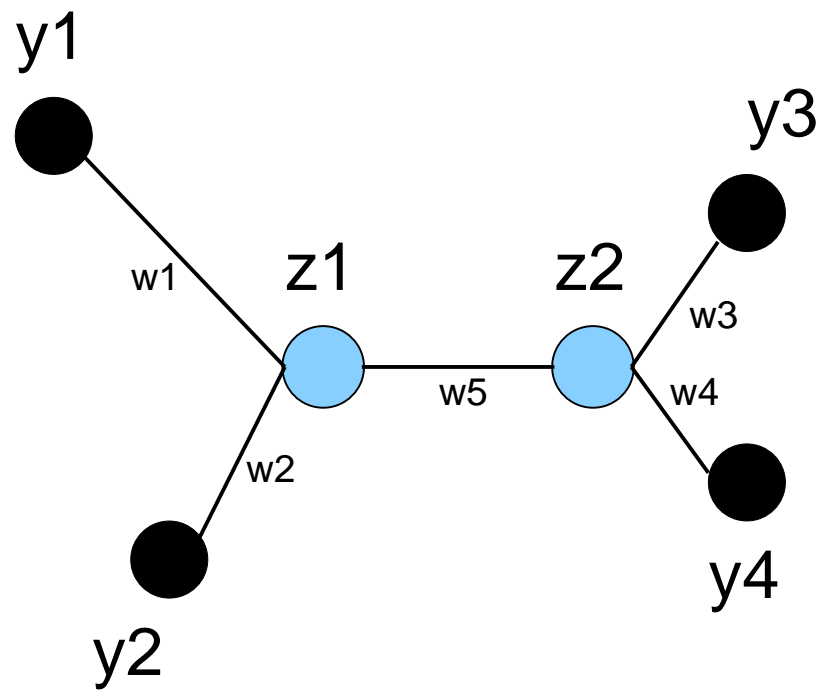


--> Topology
 --> Branch lengths

Mutation probabilities

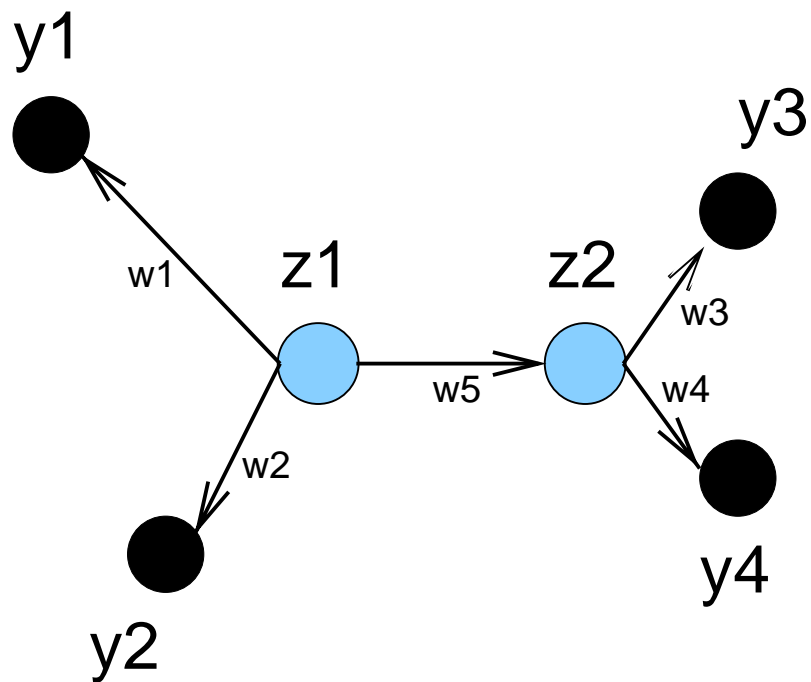


Phylogenetic tree as an undirected graph



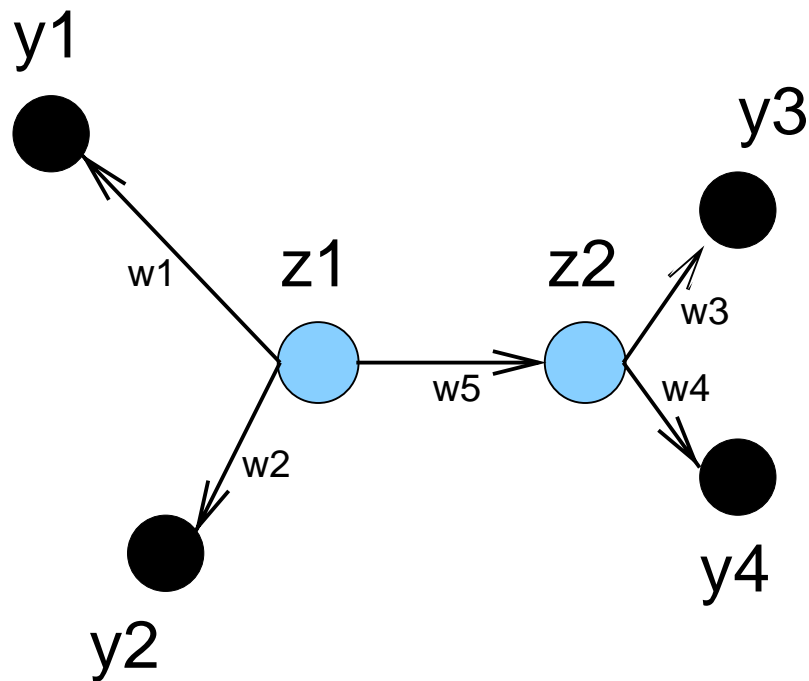
$$P(y_1, y_2, y_3, y_4, z_1, z_2 | \mathbf{w})$$

Phylogenetic tree as a directed graph



$$P(y_1, y_2, y_3, y_4, z_1, z_2 | \mathbf{w})$$

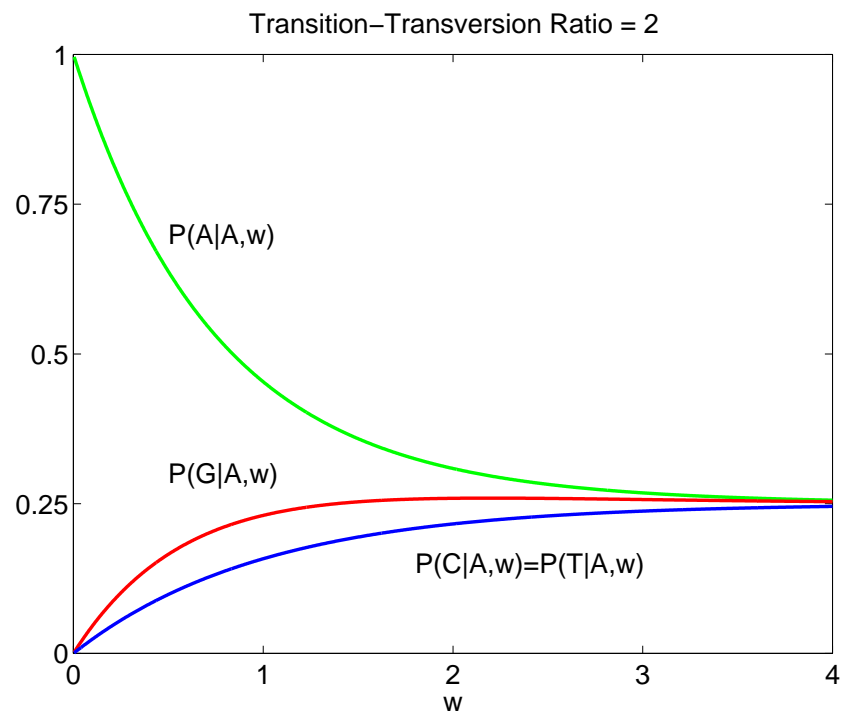
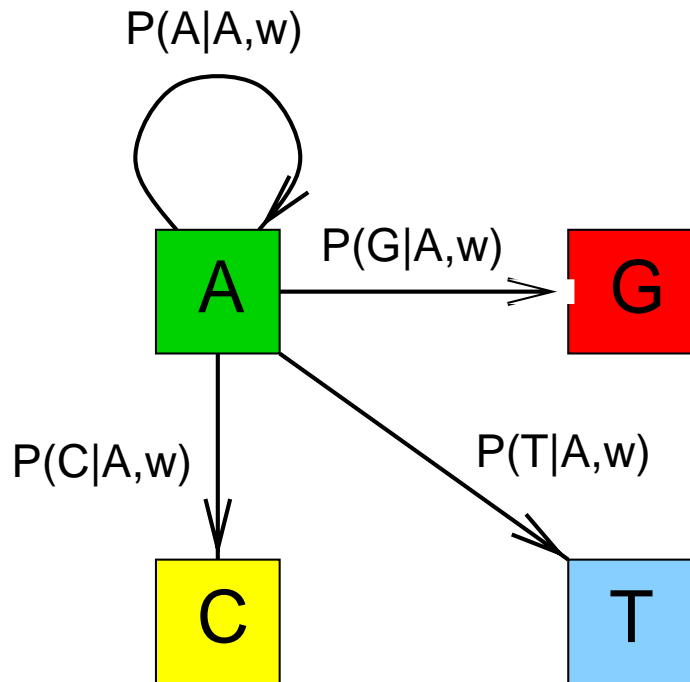
Phylogenetic tree as a directed graph



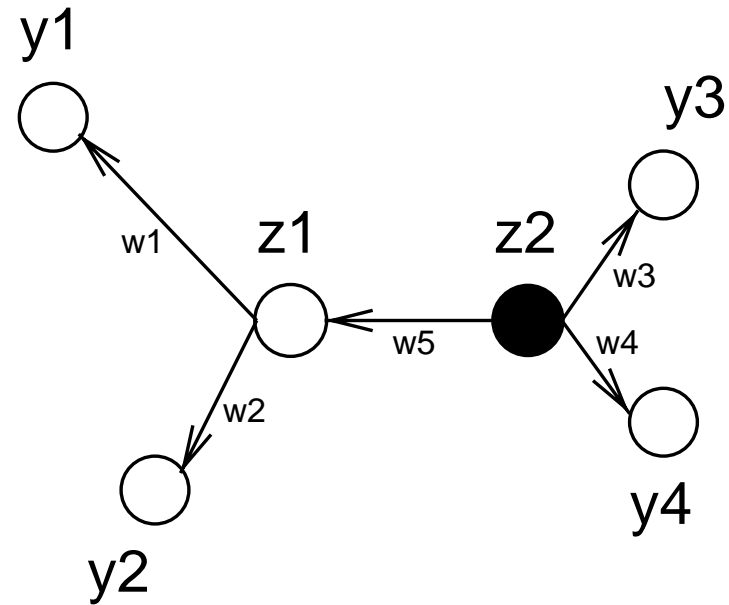
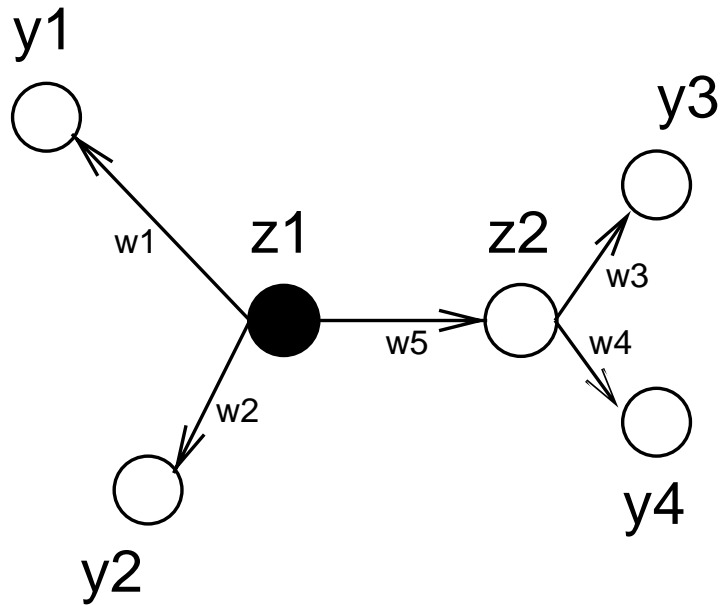
$$P(y_1, y_2, y_3, y_4, z_1, z_2 | \mathbf{w})$$

$$= P(y_1 | z_1, w_1) P(y_2 | z_1, w_2) P(z_2 | z_1, w_5) P(y_3 | z_2, w_3) P(y_4 | z_2, w_4) P(z_1)$$

Mutation probabilities



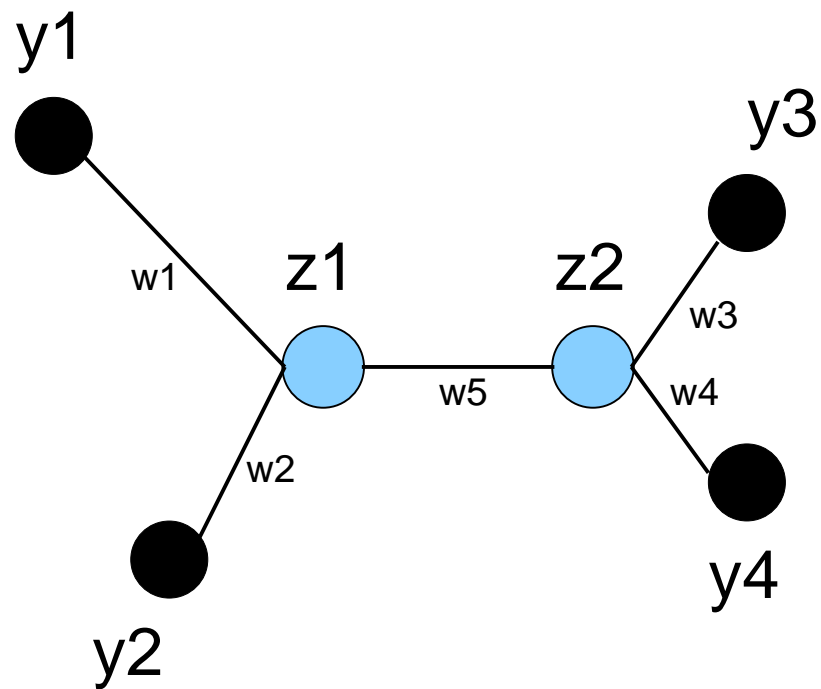
Different directed graphs



Left : $P(y_1, y_2, y_3, y_4, z_1, z_2 | \mathbf{w}) =$
 $P(y_1 | z_1, w_1) P(y_2 | z_1, w_2) P(y_3 | z_2, w_3) P(y_4 | z_2, w_4) P(z_2 | z_1, w_5) P(z_1)$

Right : $P(y_1, y_2, y_3, y_4, z_1, z_2 | \mathbf{w}) =$
 $P(y_1 | z_1, w_1) P(y_2 | z_1, w_2) P(y_3 | z_2, w_3) P(y_4 | z_2, w_4) P(z_1 | z_2, w_5) P(z_2)$

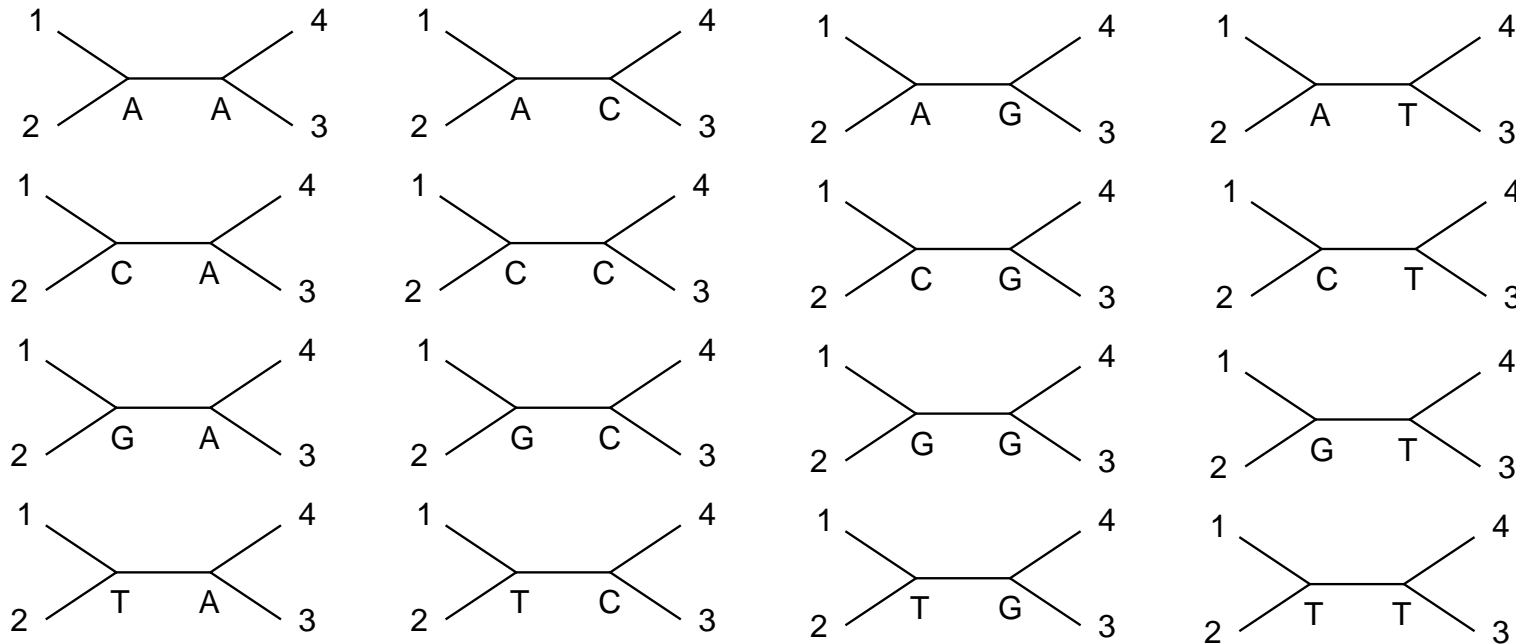
Expansion of the joint probability



$$P(y_1, y_2, y_3, y_4, z_1, z_2 | \mathbf{w})$$

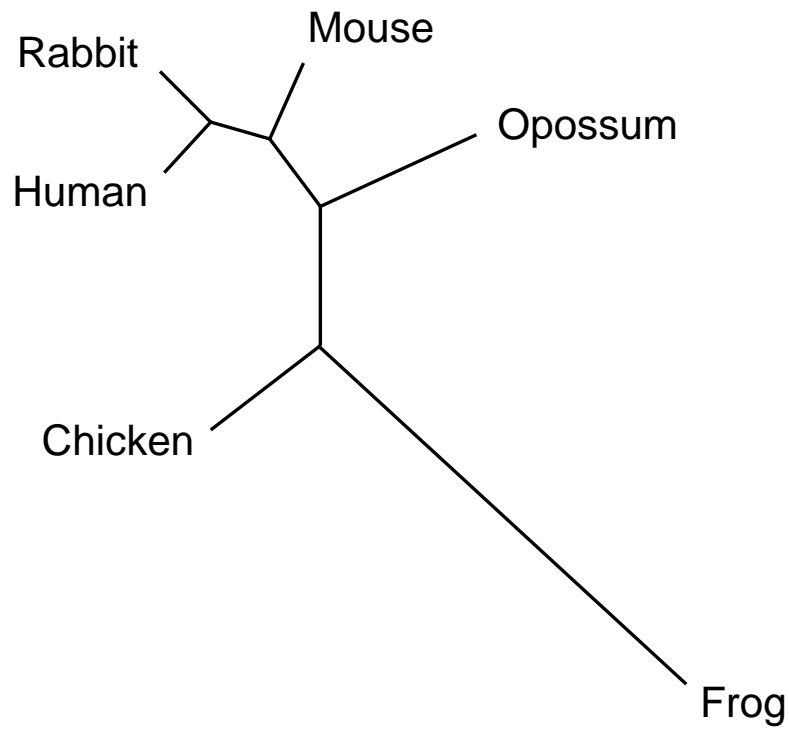
$$= P(y_1 | z_1, w_1) P(y_2 | z_1, w_2) P(z_2 | z_1, w_5) P(y_3 | z_2, w_3) P(y_4 | z_2, w_4) P(z_1)$$

Marginalisation



$$P(y_1, y_2, y_3, y_4 | \mathbf{w}) = \sum_{z_1} \sum_{z_2} P(y_1, y_2, y_3, y_4, z_1, z_2 | \mathbf{w})$$

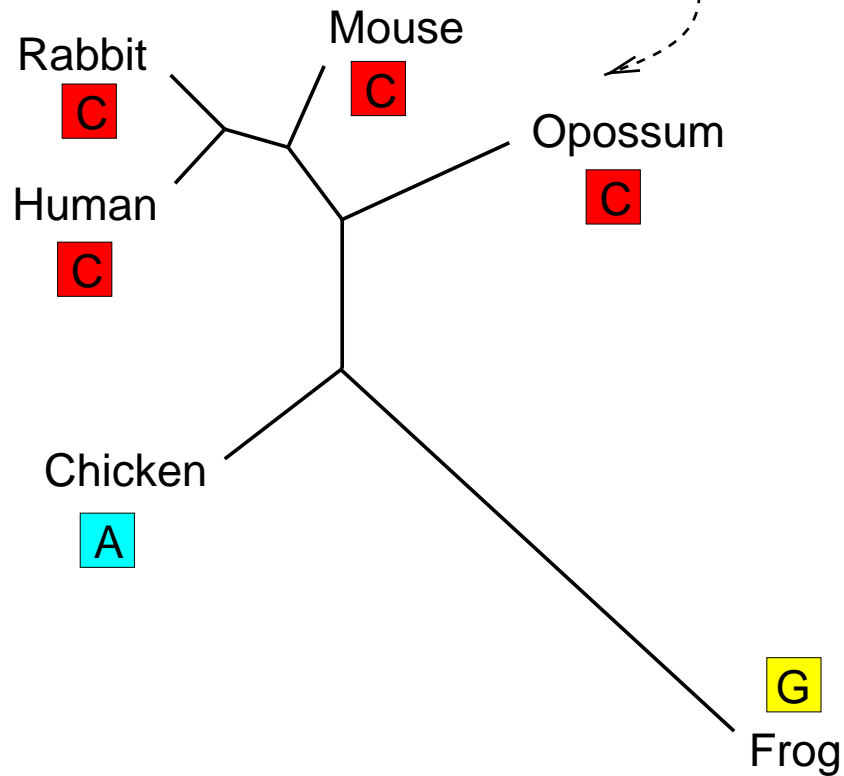
Frog	G	C	T	T	G	A	C	T	T	C	T	G	A	G	G	T	T
Chicken	G	C	G	T	A	A	C	T	T	C	A	C	A	T	G	A	T
Human	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T
Rabbit	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T
Mouse	G	C	G	T	C	A	C	T	T	G	A	C	A	G	G	C	T
Opossum	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T



--> Topology
 --> Branch lengths

∇

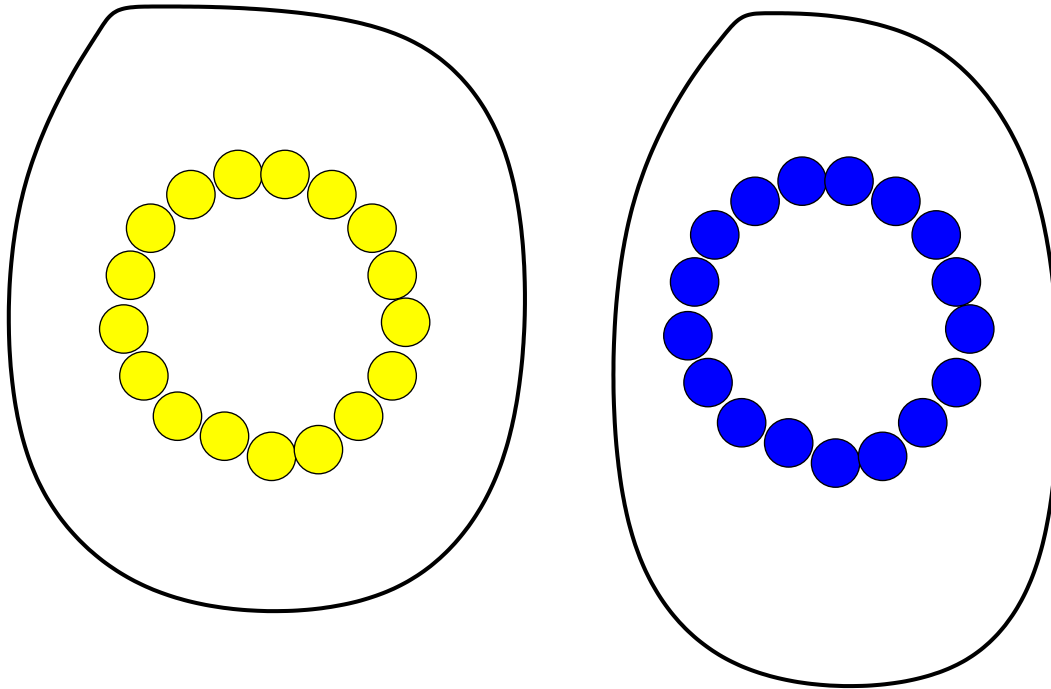
Frog	G	C	T	T	G	A	C	T	T	C	T	G	A	G	G	T	T
Chicken	G	C	G	T	A	A	C	T	T	C	A	C	A	T	G	A	T
Human	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T
Rabbit	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T
Mouse	G	C	G	T	C	A	C	T	T	G	A	C	A	G	G	C	T
Opossum	G	C	G	T	C	A	C	T	T	G	A	G	A	C	G	C	T



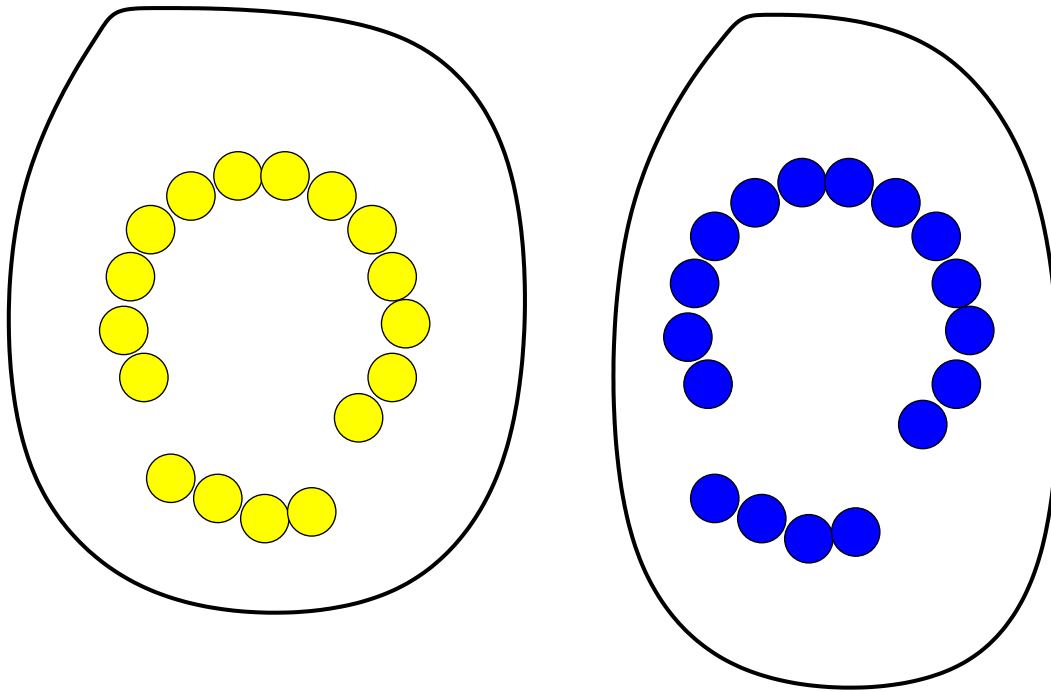
--> Likelihood

Topology
Branch lengths

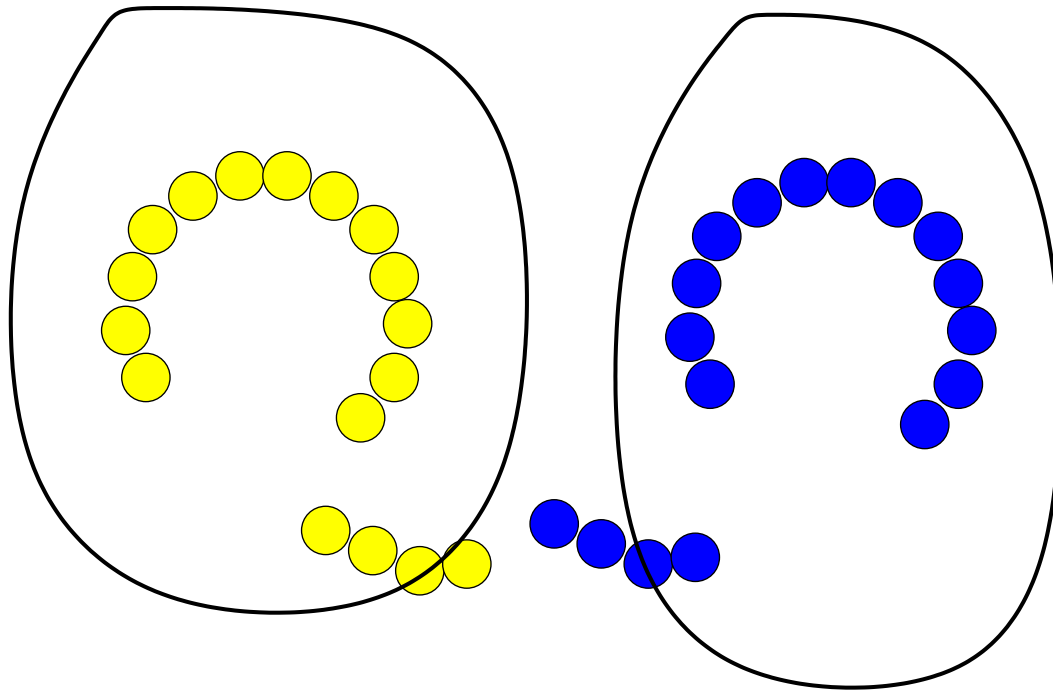
Recombination



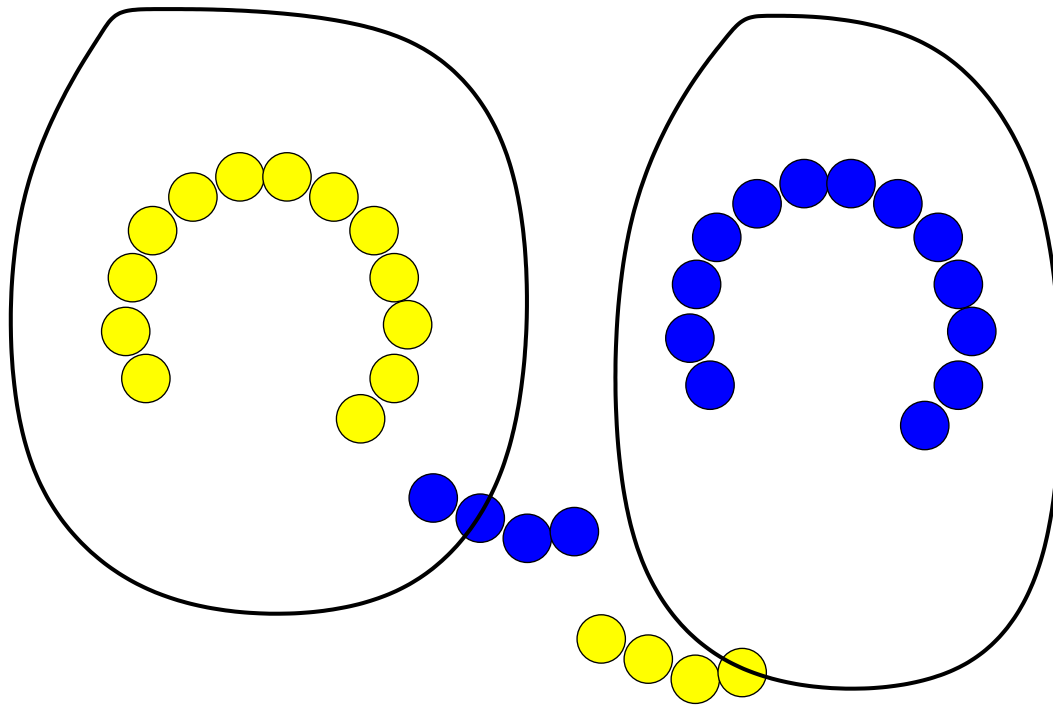
Recombination



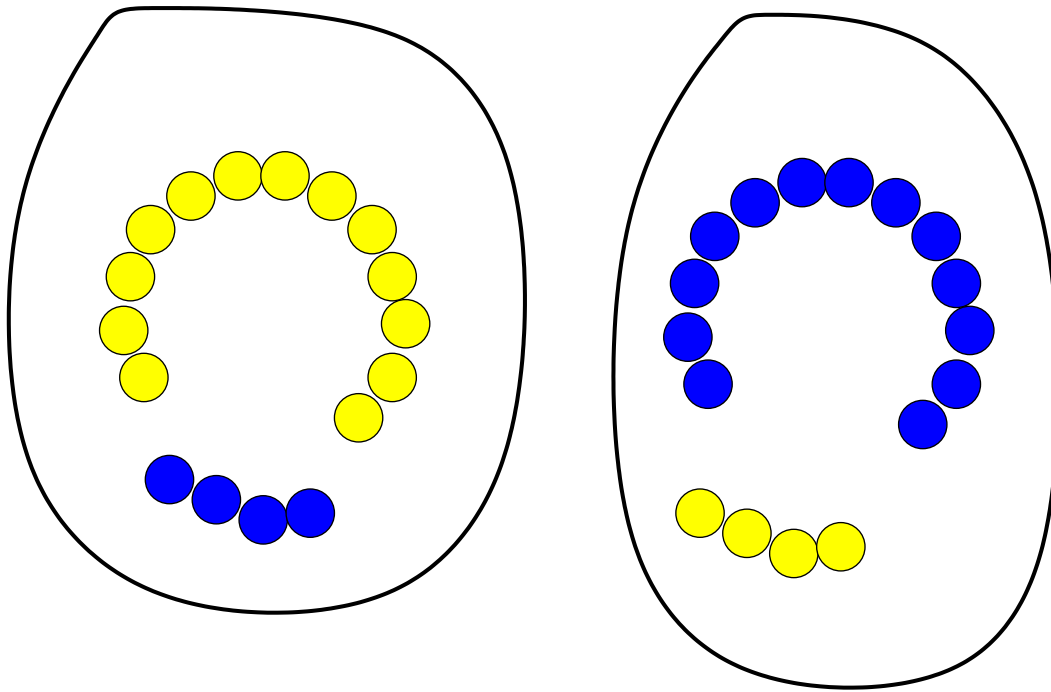
Recombination



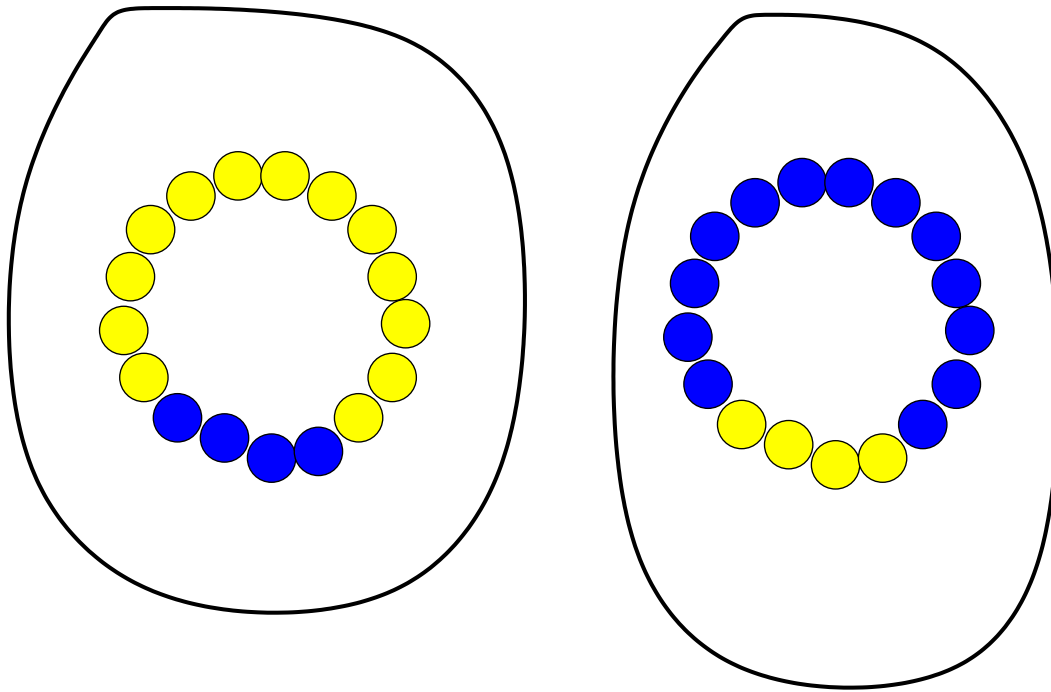
Recombination



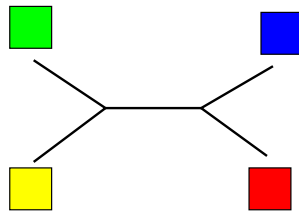
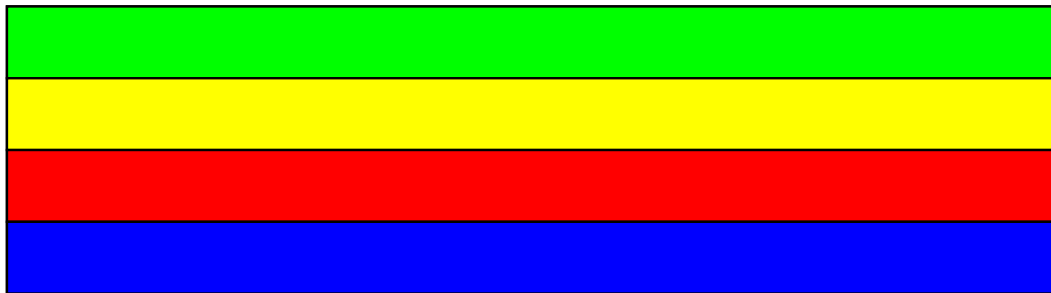
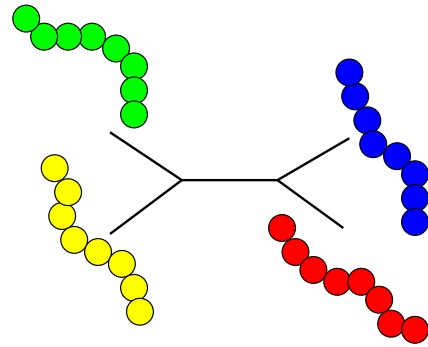
Recombination



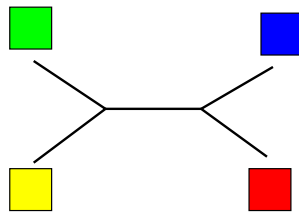
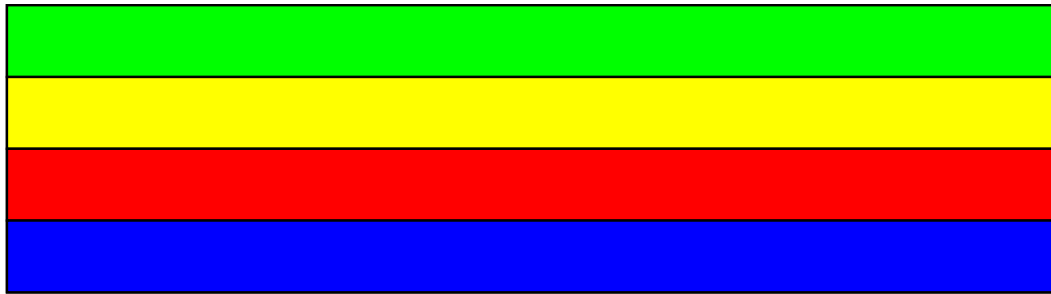
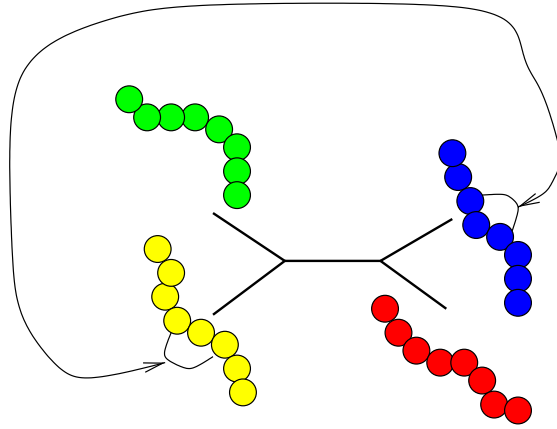
Recombination



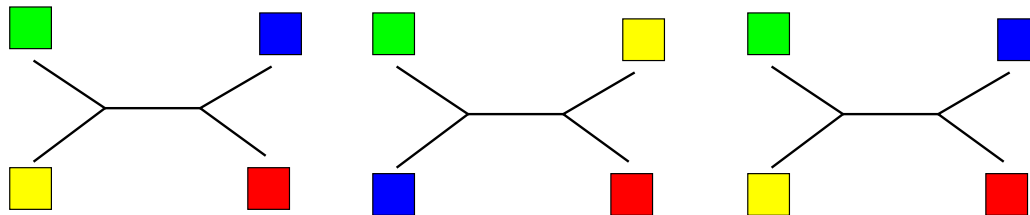
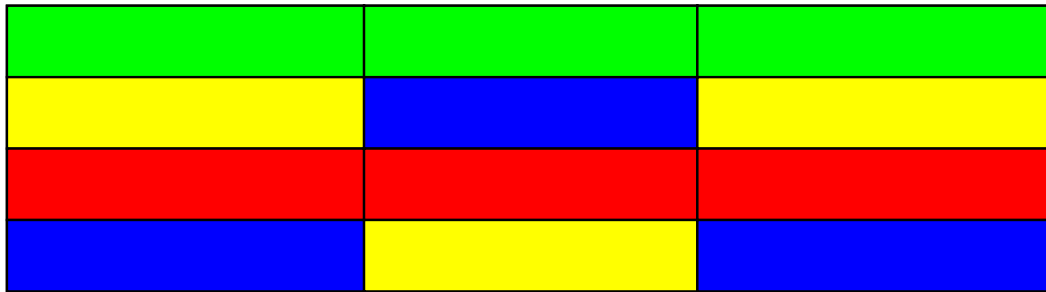
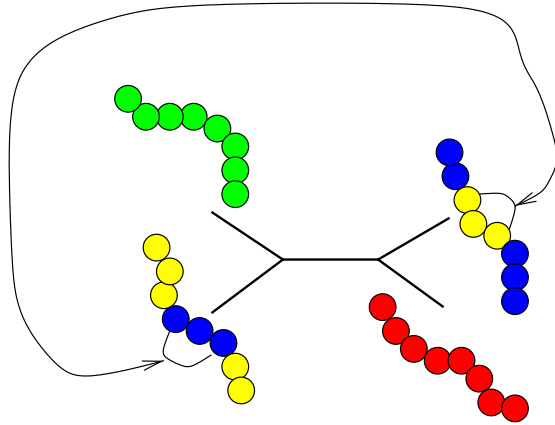
Recombination



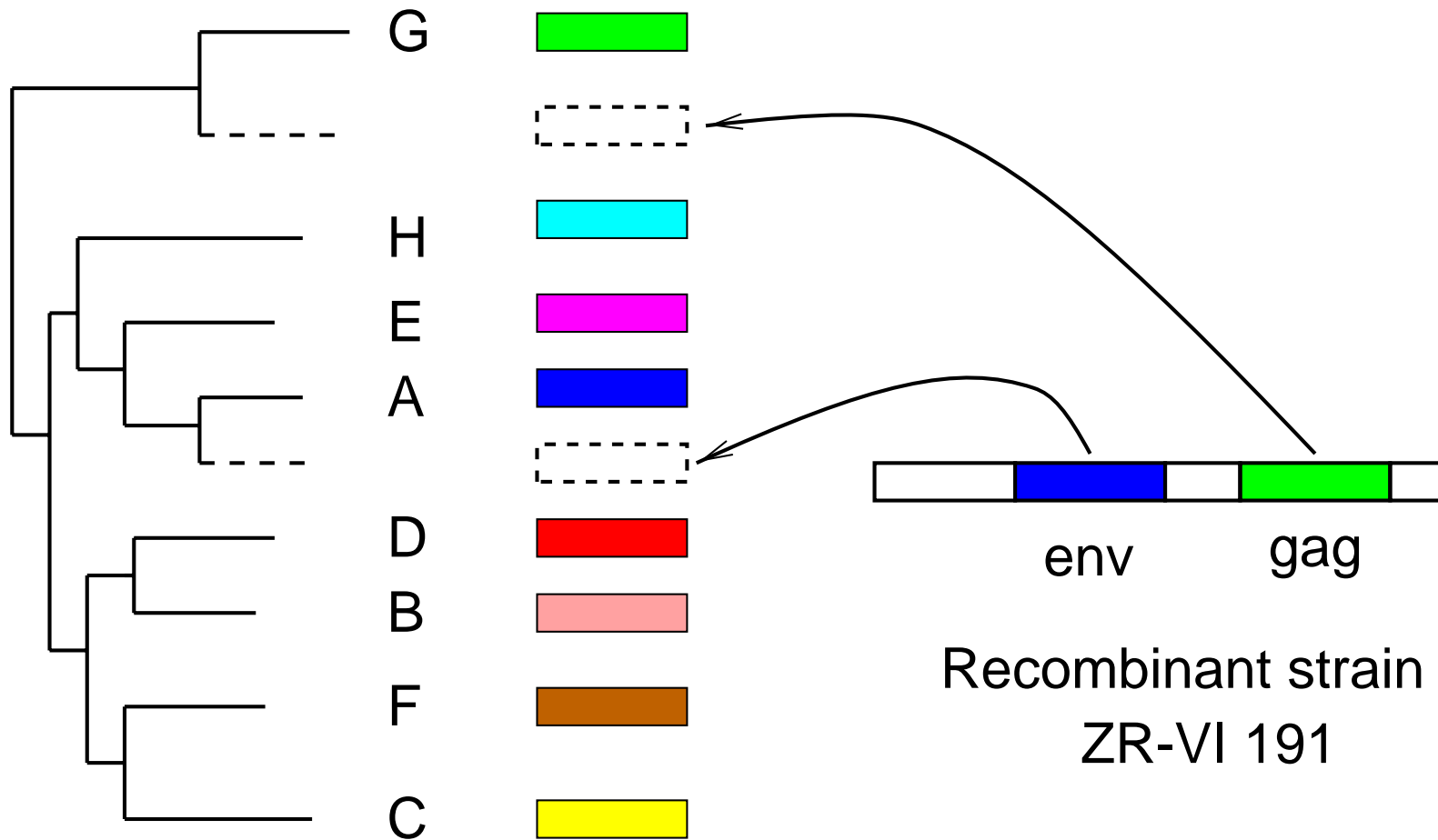
Recombination



Recombination



Recombination in HIV 1



Detecting recombination with window methods

- Slide a **window** across the alignment.
- Look for **subregions** that are **significantly different** from the rest of the alignment.

Detecting recombination with window methods

- **DSS (TOPAL)**
McGuire, Wright, Prentice (Mol. Biol. Evol. 14)
- **PDM**
Husmeier, Wright (Bioinformatics 18)
- **Pruned PDM**
Work in progress

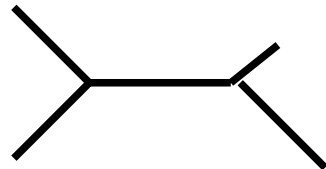
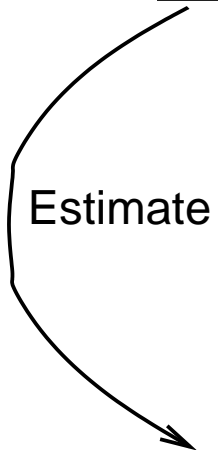
Detecting recombination with window methods

- **DSS (TOPAL)**
McGuire, Wright, Prentice (Mol. Biol. Evol. 14)
- **PDM**
Husmeier, Wright (Bioinformatics 18)
- **Pruned PDM**
Work in progress

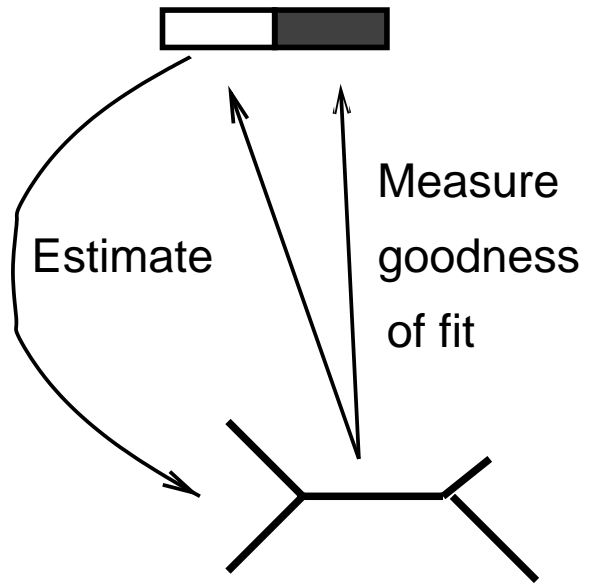
DSS (McGuire & Wright, 1997)



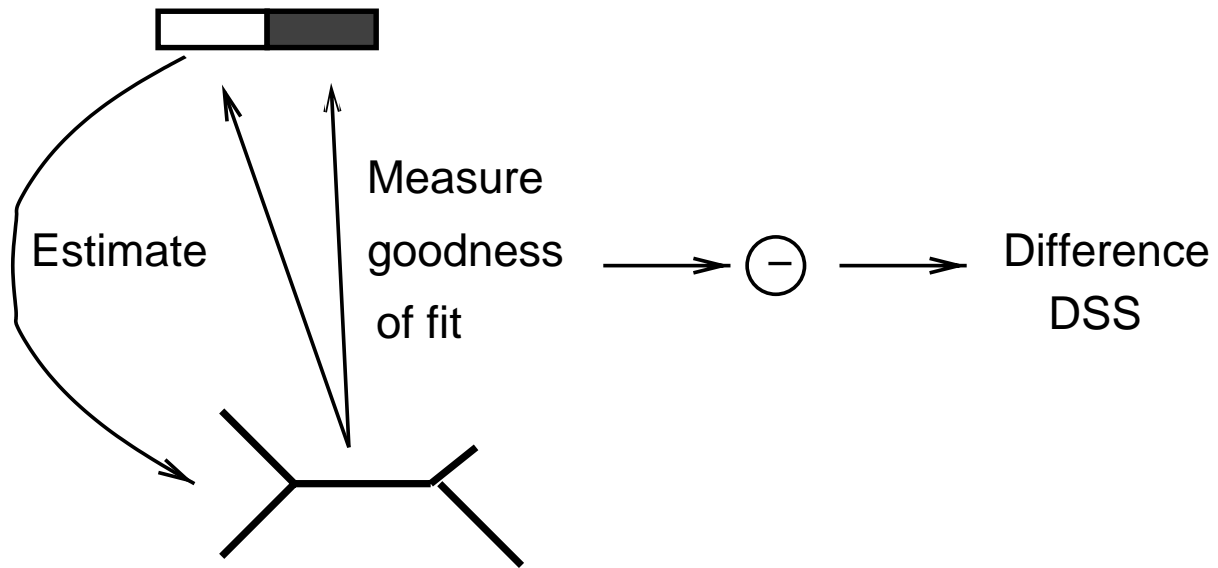
DSS (McGuire & Wright, 1997)



DSS (McGuire & Wright, 1997)



DSS (McGuire & Wright, 1997)



DSS (McGuire & Wright, 1997)



small

DSS (McGuire & Wright, 1997)



small



large

DSS (McGuire & Wright, 1997)



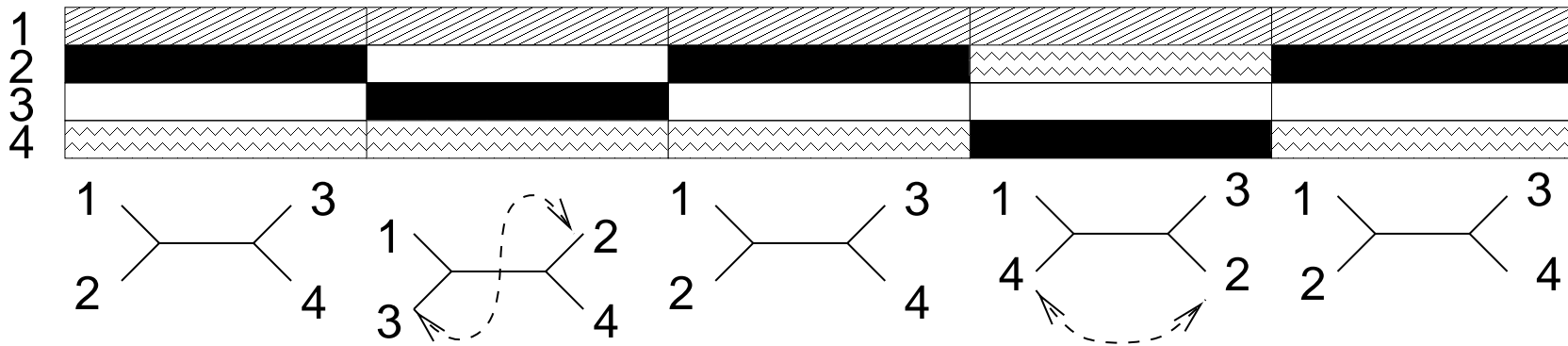
small



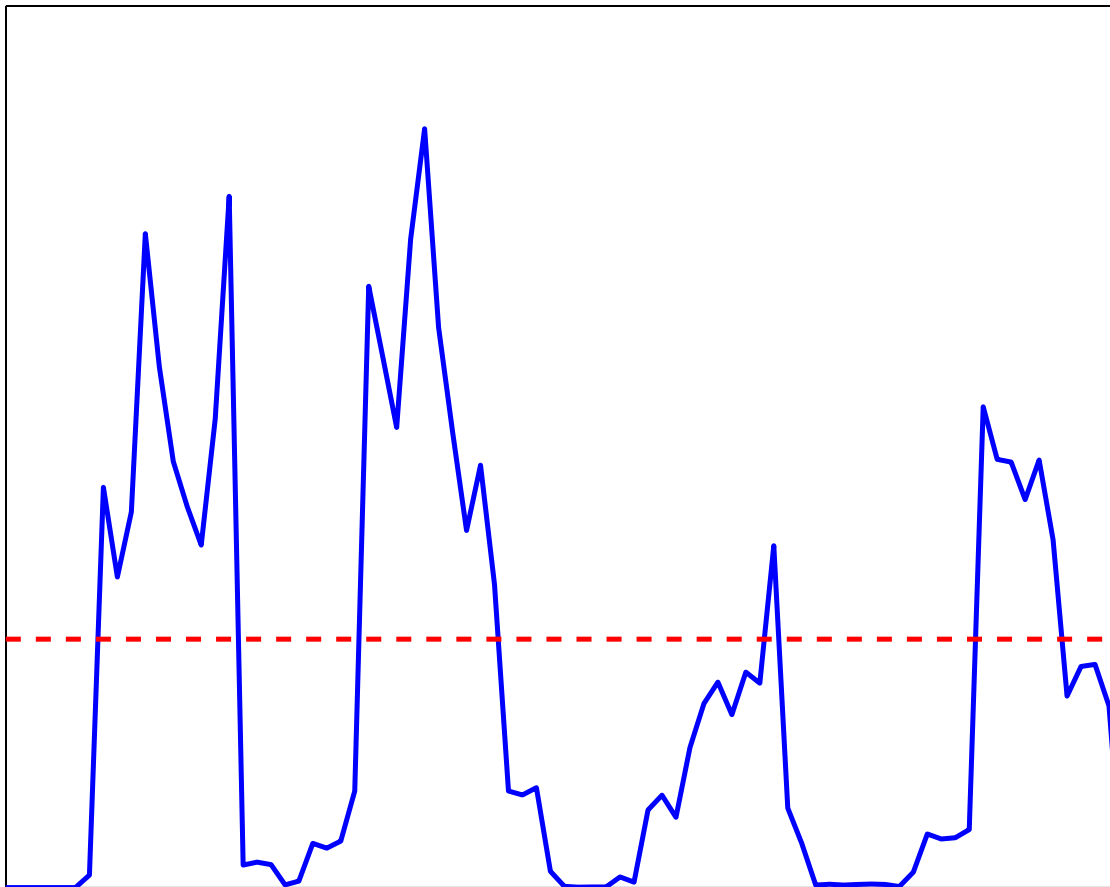
large

- Detect **significant peaks** of the DSS signal.
- Significance determined with **parametric bootstrapping**.

Example



Example: DSS, window size=200

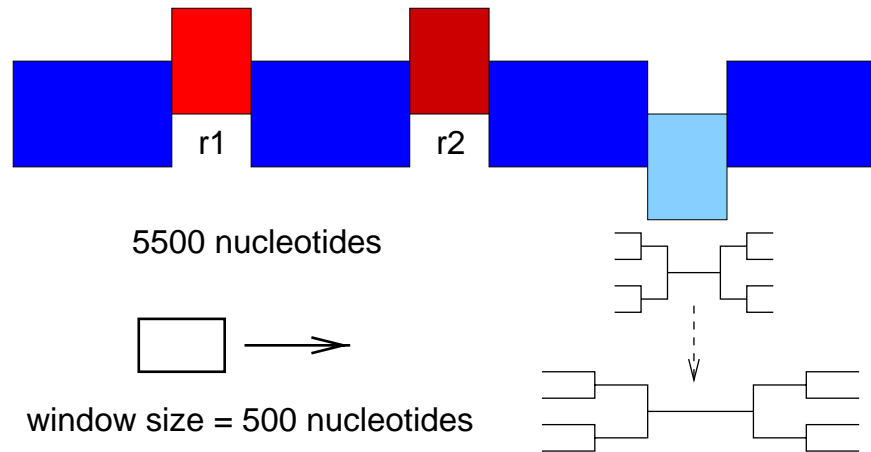
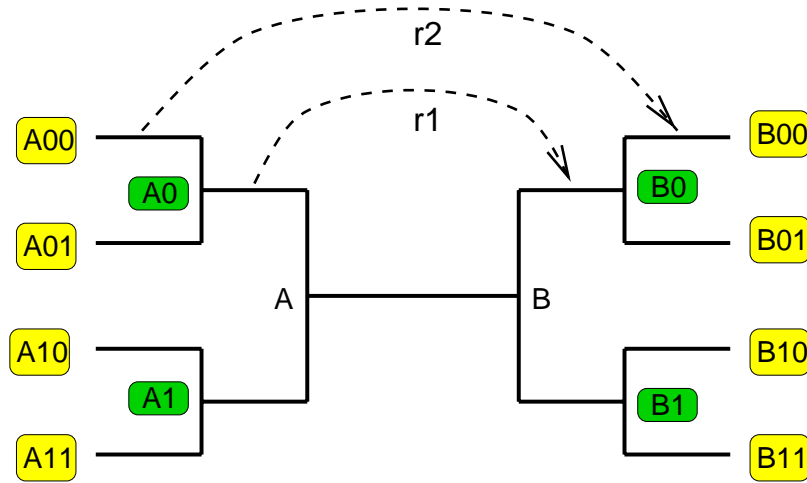


Problems with DSS

Problems with DSS

- Intrinsic **uncertainty** of reference model due to finite window size.
- We need to distinguish between **recombination** and **rate variation**.

Recombination versus rate variation

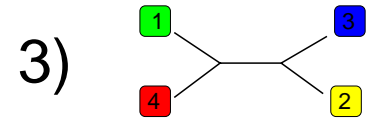
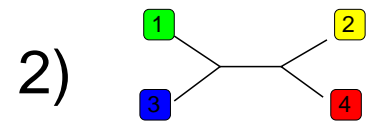
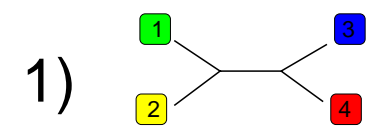
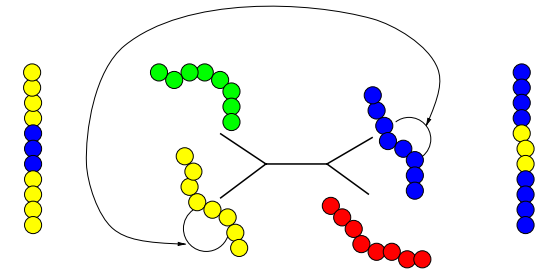
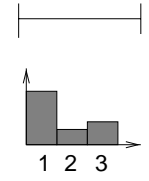
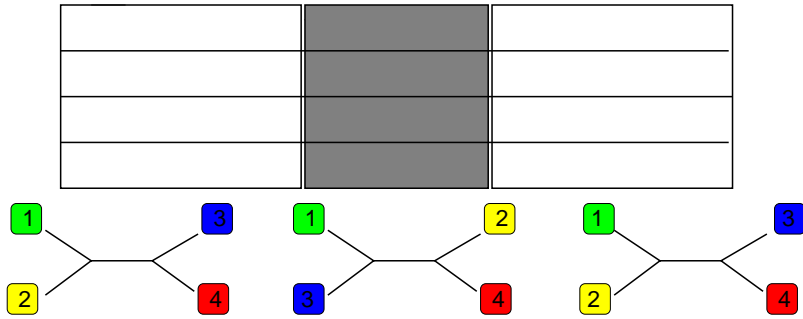


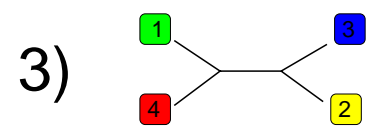
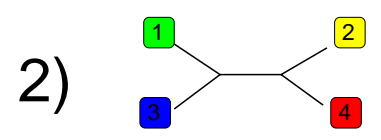
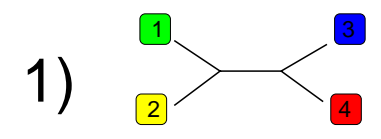
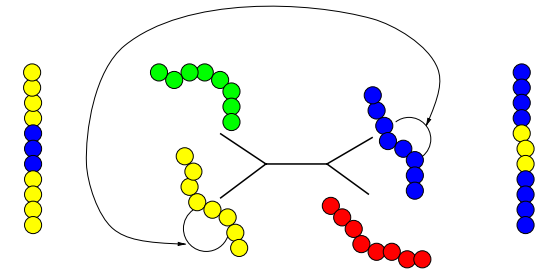
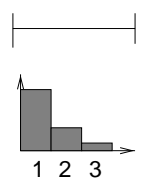
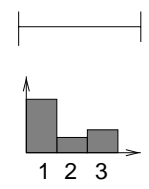
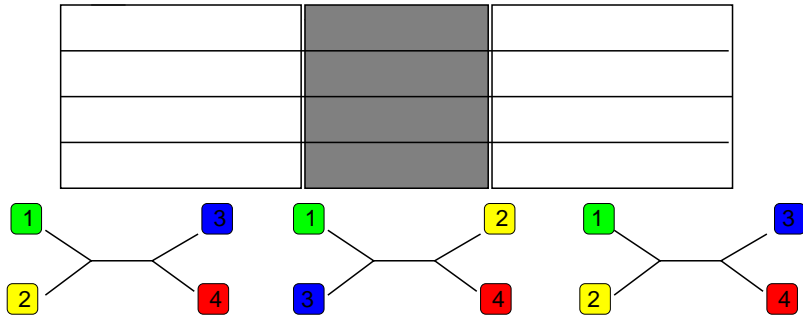
Objective

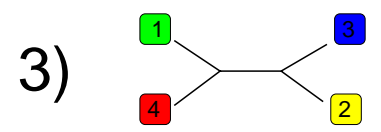
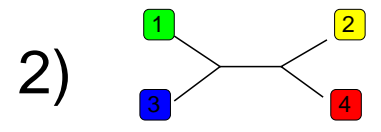
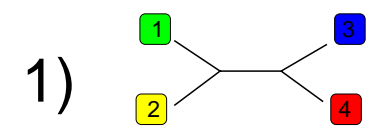
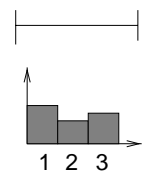
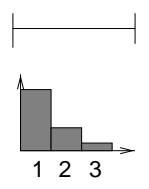
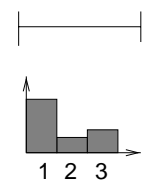
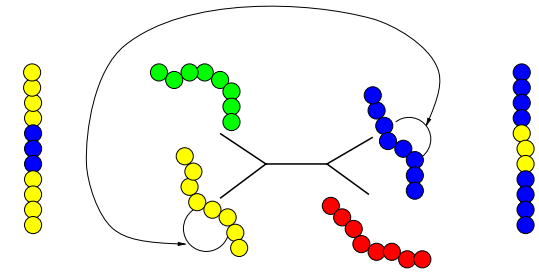
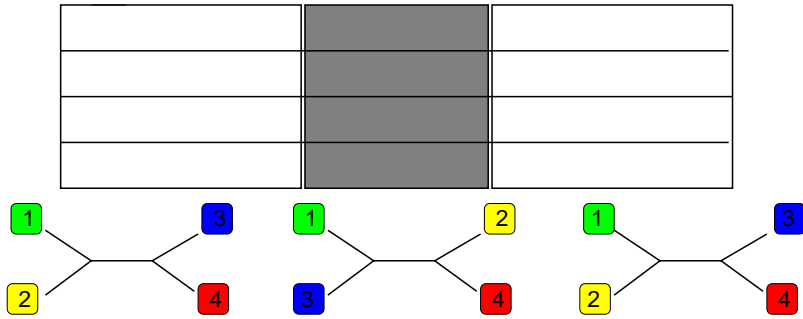
- Capture **intrinsic uncertainty** due to **finite window size**.
- Focus on **topology changes** to distinguish between **recombination** and **rate variation**.

Detecting recombination with window methods

- **DSS (TOPAL)**
McGuire, Wright, Prentice (Mol. Biol. Evol. 14)
- **PDM**
Husmeier, Wright (Bioinformatics 18)
- **Pruned PDM**
Work in progress

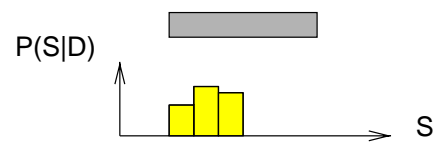
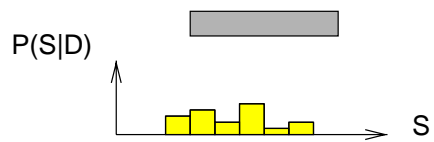
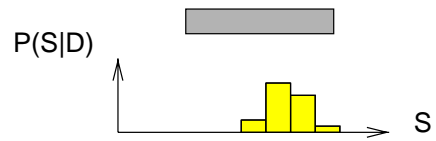
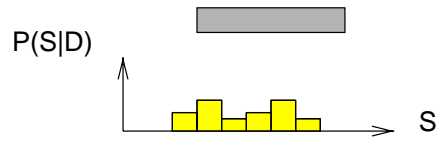
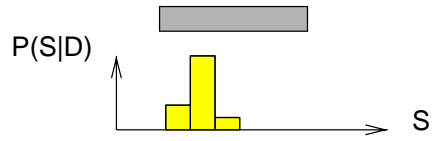




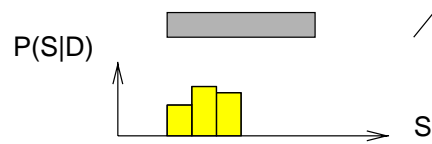
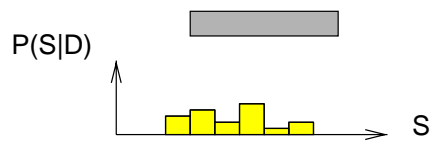
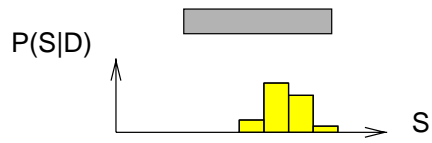
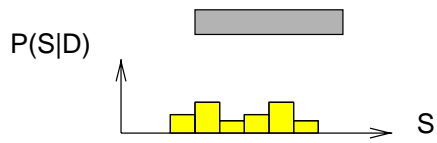
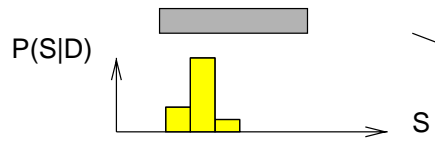


DNA sequence alignment

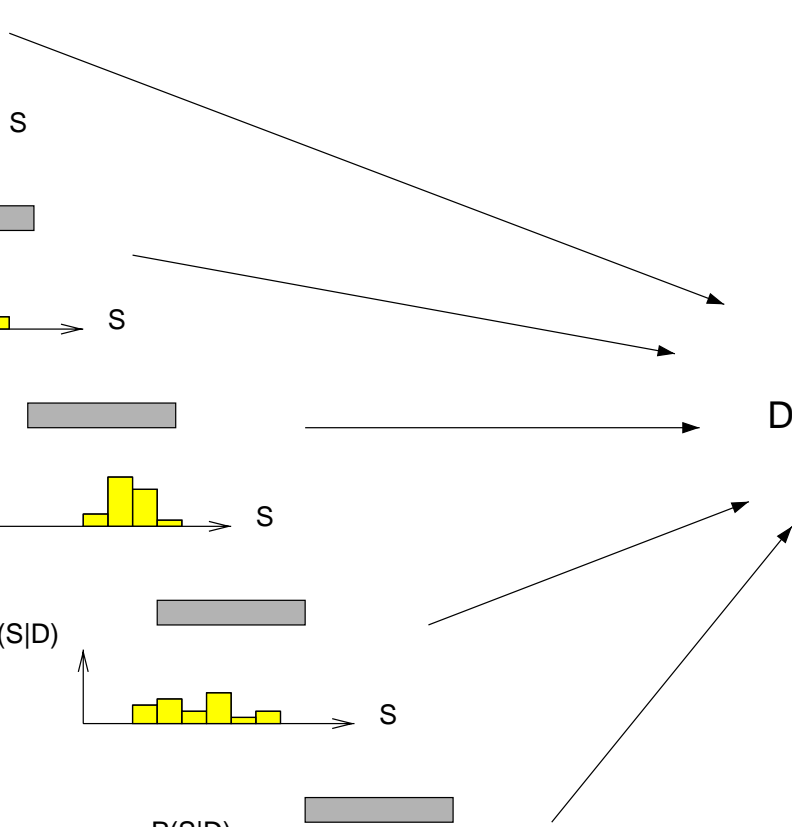
DNA sequence alignment



DNA sequence alignment



Difference



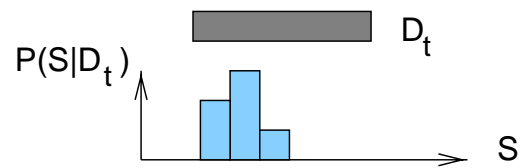
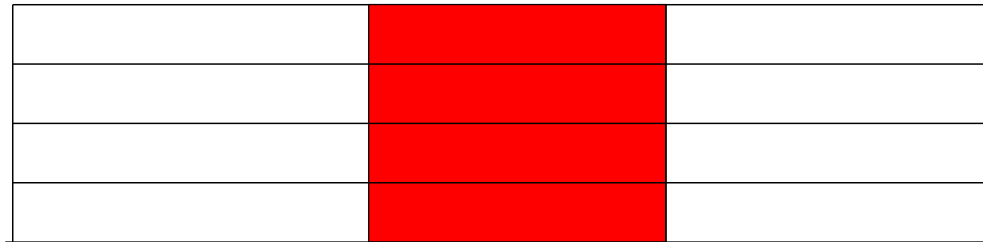
PDM method

- Posterior probability distribution over tree topologies
- Probabilistic divergence measure (PDM)
- Significance estimation

PDM method

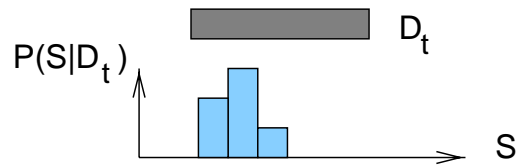
- Posterior probability distribution over tree topologies
- Probabilistic divergence measure (PDM)
- Significance estimation

Marginal posterior distribution of tree topologies with MCMC



$$P(S|\mathcal{D}_t) = \int P(S, \mathbf{w}|\mathcal{D}_t) d\mathbf{w}$$

Marginal posterior distribution of tree topologies with MCMC



$$P(S|\mathcal{D}_t) = \int P(S, \mathbf{w}|\mathcal{D}_t) d\mathbf{w}$$

$$\text{MCMC} \longrightarrow \text{Sample : } \{S_{ti}, \mathbf{w}_{ti}\}_{i=1}^N$$

$$P(S, \mathbf{w}|\mathcal{D}_t) \approx \frac{1}{N} \sum_{i=1}^N \delta_{S, S_{ti}} \delta(\mathbf{w} - \mathbf{w}_{ti})$$

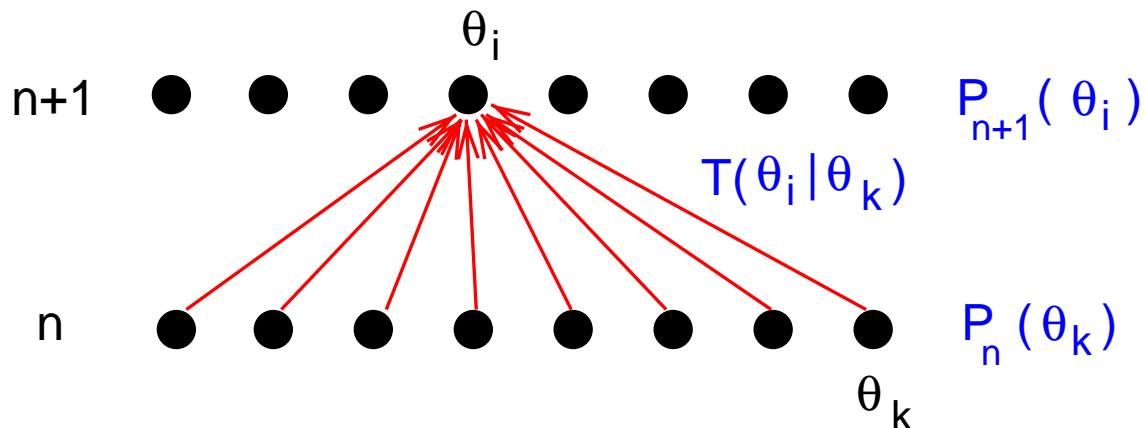
$$P(S|\mathcal{D}_t) = \frac{1}{N} \sum_{i=1}^N \delta_{S, S_{ti}} = \frac{N_S(t)}{N}$$

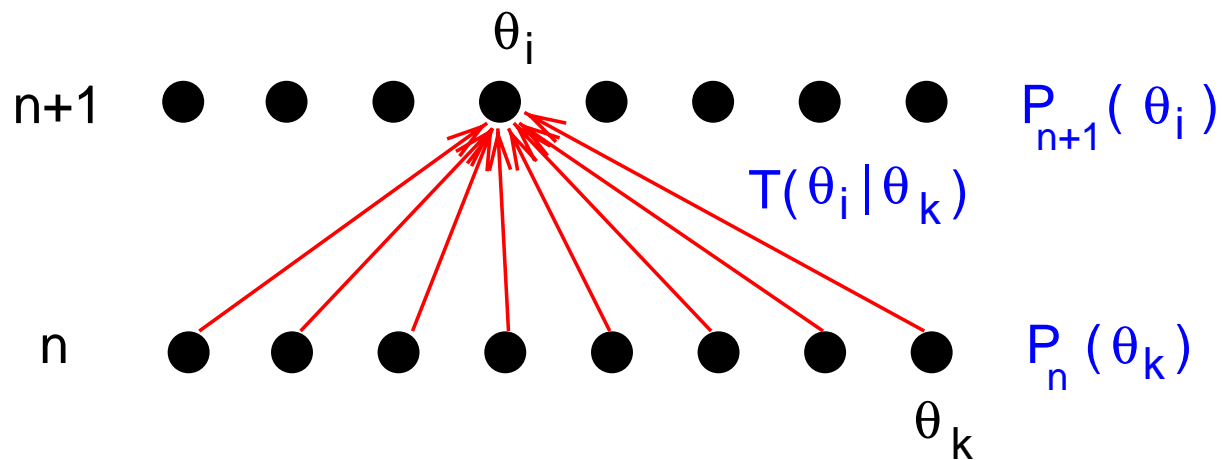
Markov chain Monte Carlo (MCMC)

- Objective: Sample from the posterior distribution

$$P(\theta|D) = \frac{P(D|\theta)P(\theta)}{P(D)} = \frac{P(D|\theta)P(\theta)}{\int P(D|\theta)P(\theta)d\theta}$$

- Direct approach intractable due to $\int P(D|\theta)P(\theta)d\theta$
- Devise a Markov chain $P_{n+1}(\theta_i) = \sum_k T(\theta_i|\theta_k)P_n(\theta_k)$ that converges in distribution to $P(\theta|D)$: $P_n(\theta) \rightarrow P(\theta|D)$





- Theorem: An **ergodic** Markov chain converges to its **stationary distribution** irrespective of its **initialization**.
- Stationary distribution: $P(\theta_i) = \sum_k T(\theta_i|\theta_k)P(\theta_k)$
- Design the **Markov transition matrix** $T(\theta_i|\theta_k)$ such that $P(\theta|D)$ is the stationary distribution.
- Detailed balance: $\frac{T(\theta_k|\theta_i)}{T(\theta_i|\theta_k)} = \frac{P(\theta_k|D)}{P(\theta_i|D)} = \frac{P(D|\theta_k)P(\theta_k)}{P(D|\theta_i)P(\theta_i)}$

Proof

Show that $\sum_k T(\theta_i|\theta_k)P(\theta_k|\mathcal{D}) = P(\theta_i|\mathcal{D})$

Detailed balance: $\frac{T(\theta_k|\theta_i)}{T(\theta_i|\theta_k)} = \frac{P(\theta_k|\mathcal{D})}{P(\theta_i|\mathcal{D})} \implies$

$$T(\theta_k|\theta_i)P(\theta_i|\mathcal{D}) = T(\theta_i|\theta_k)P(\theta_k|\mathcal{D}) \implies$$

$$\begin{aligned} \sum_k T(\theta_i|\theta_k)P(\theta_k|\mathcal{D}) &= \sum_k T(\theta_k|\theta_i)P(\theta_i|\mathcal{D}) = \\ &P(\theta_i|\mathcal{D}) \sum_k T(\theta_k|\theta_i) = P(\theta_i|\mathcal{D}) \end{aligned}$$

Metropolis-Hastings algorithm

$$\frac{T(\theta_k|\theta_i)}{T(\theta_i|\theta_k)} = \frac{P(D|\theta_k)P(\theta_k)}{P(D|\theta_i)P(\theta_i)}$$

Transition Probability = Proposal Probability \times Acceptance Probability

$$T(\theta_k|\theta_i) = q(\theta_k|\theta_i)a(\theta_k|\theta_i)$$

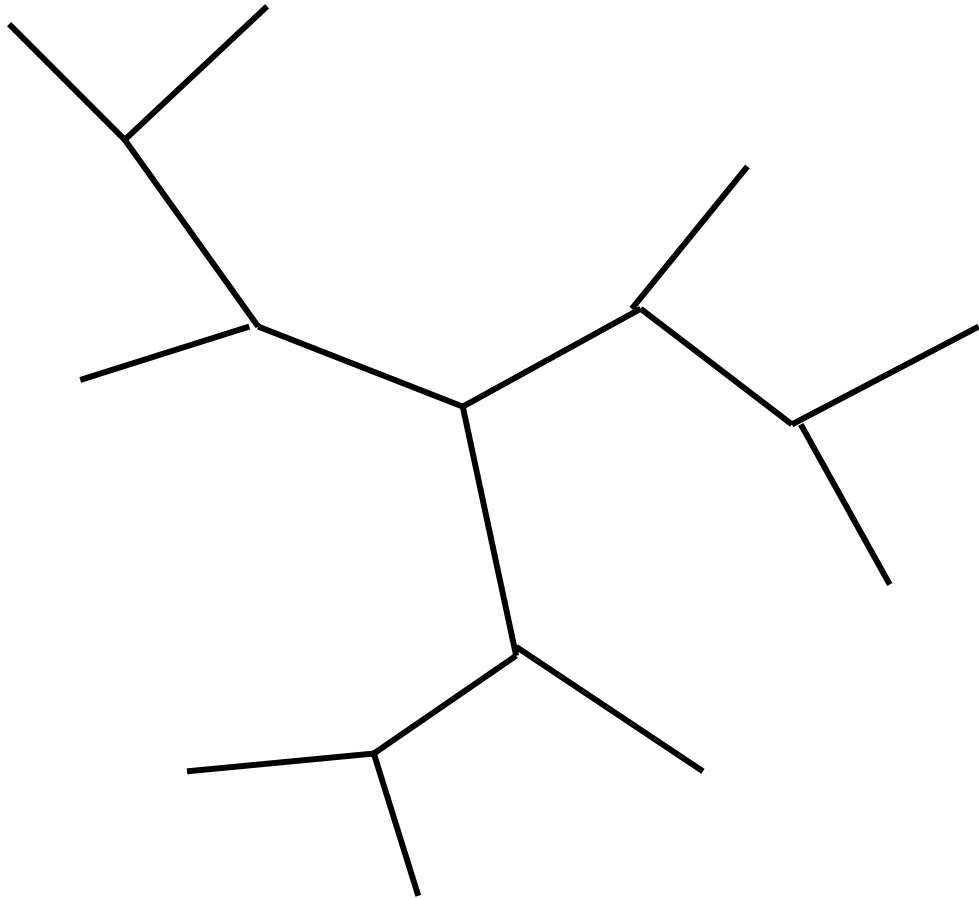
Acceptance Probabilities:

$$\frac{a(\theta_k|\theta_i)}{a(\theta_i|\theta_k)} = \frac{P(D|\theta_k)P(\theta_k)q(\theta_i|\theta_k)}{P(D|\theta_i)P(\theta_i)q(\theta_k|\theta_i)}$$

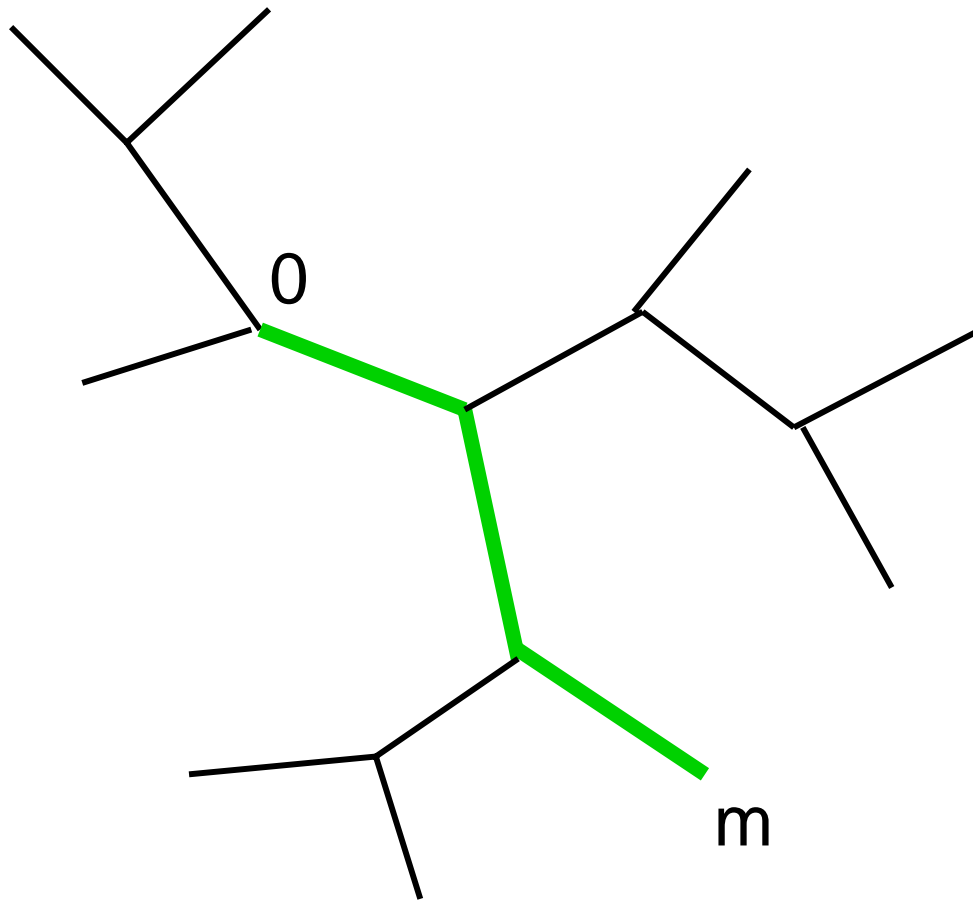
$$a(\theta_k|\theta_i) = \min \left\{ \frac{P(D|\theta_k)P(\theta_k)q(\theta_i|\theta_k)}{P(D|\theta_i)P(\theta_i)q(\theta_k|\theta_i)}, 1 \right\}$$

MCMC Implementation

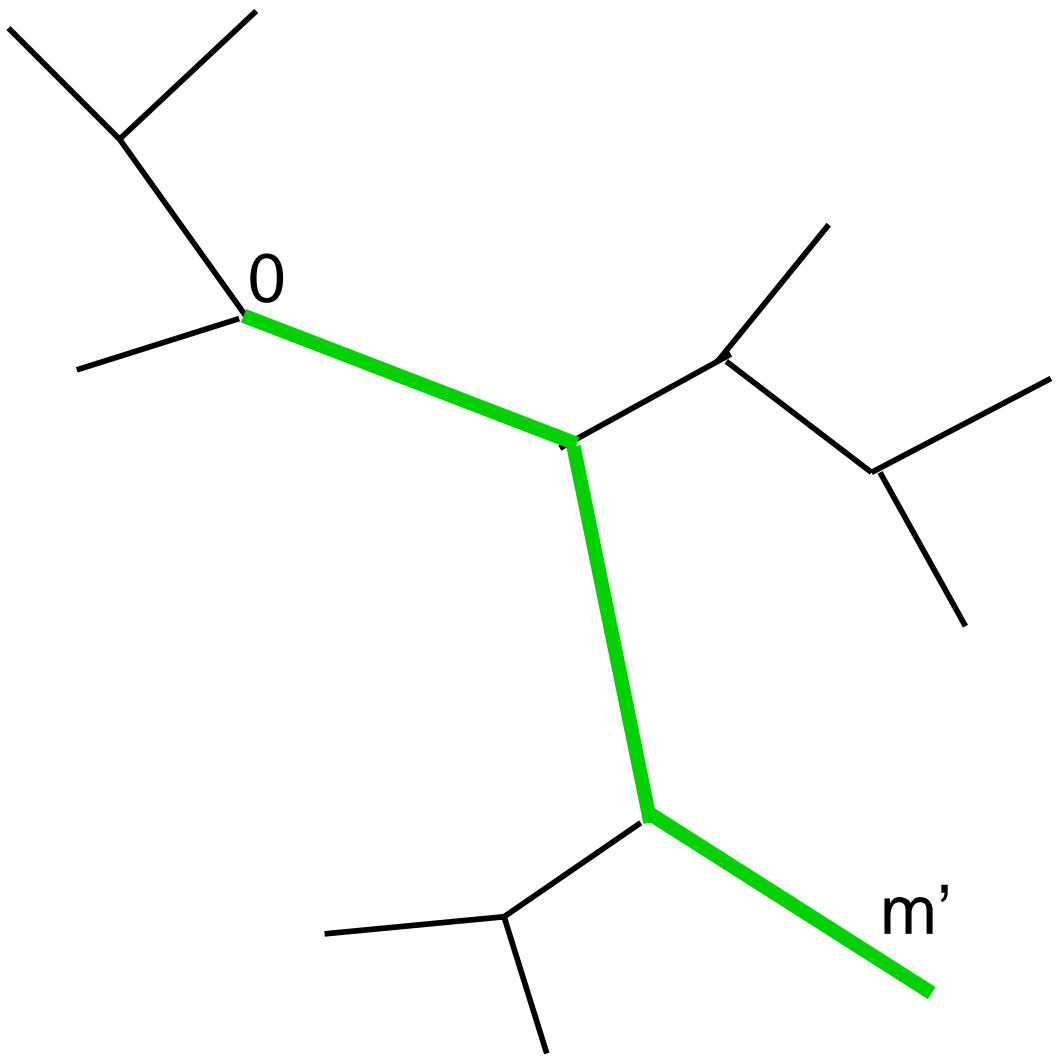
- B. Larget, D. L. Simon (1999)
Molecular Biology and Evolution 6 (16), 750-759
- BAMBE:
Bayesian Analysis in Molecular Biology and Evolution
<http://www.mathcs.duq.edu/larget/bambe.html>



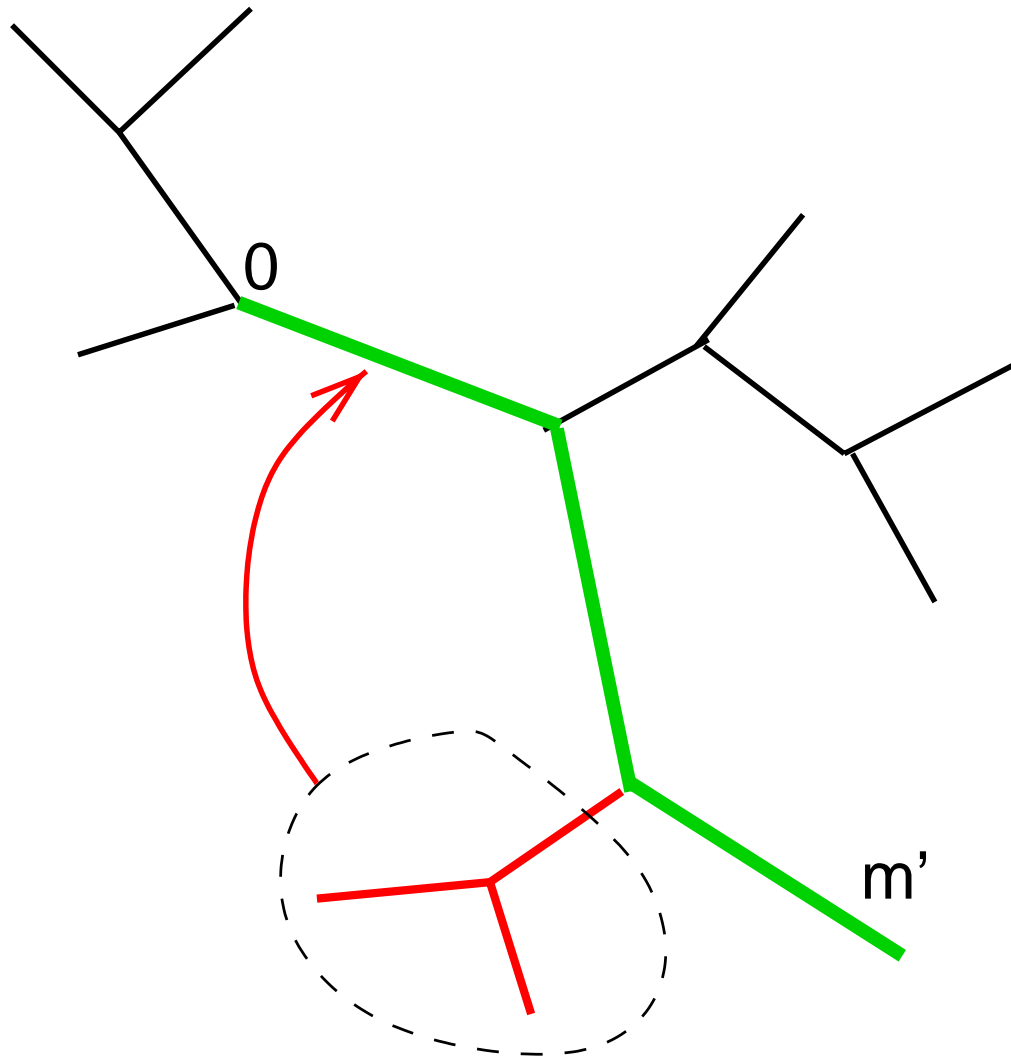
..

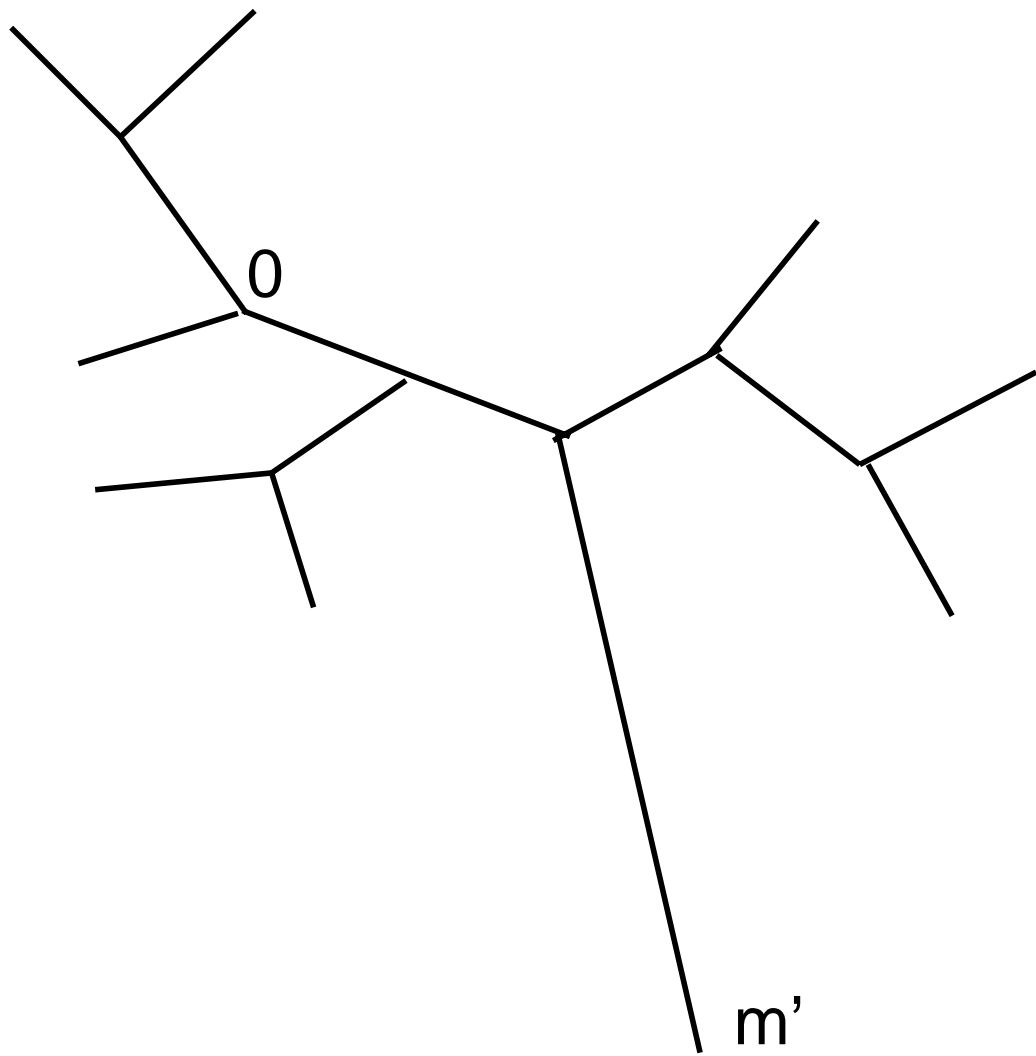


..



..





..

Acceptance probability

=

ratio of prior probabilities ×
likelihood ratio ×
inverse ratio of proposal probabilities

Acceptance probability

=

ratio of prior probabilities ×

likelihood ratio ×

inverse ratio of proposal probabilities

Prior → uniform

Likelihood → message passing (Felsenstein's pruning algorithm)

Hastings factor = inverse ratio of proposal probabilities

Hastings factor

- Old length: m
- New length: $m' = m \exp(\lambda(U - 0.5))$
 U : Uniform random variable in $[0, 1]$
 $\lambda > 0$: Tuning parameter
- Select one of the branches with equal probabilities
- Regraft this branch; new position chosen uniformly from $[0, m']$.

What is the **Hastings ratio**: $\frac{P(\text{backward move})}{P(\text{forward move})}$?

Hastings factor

Forward move

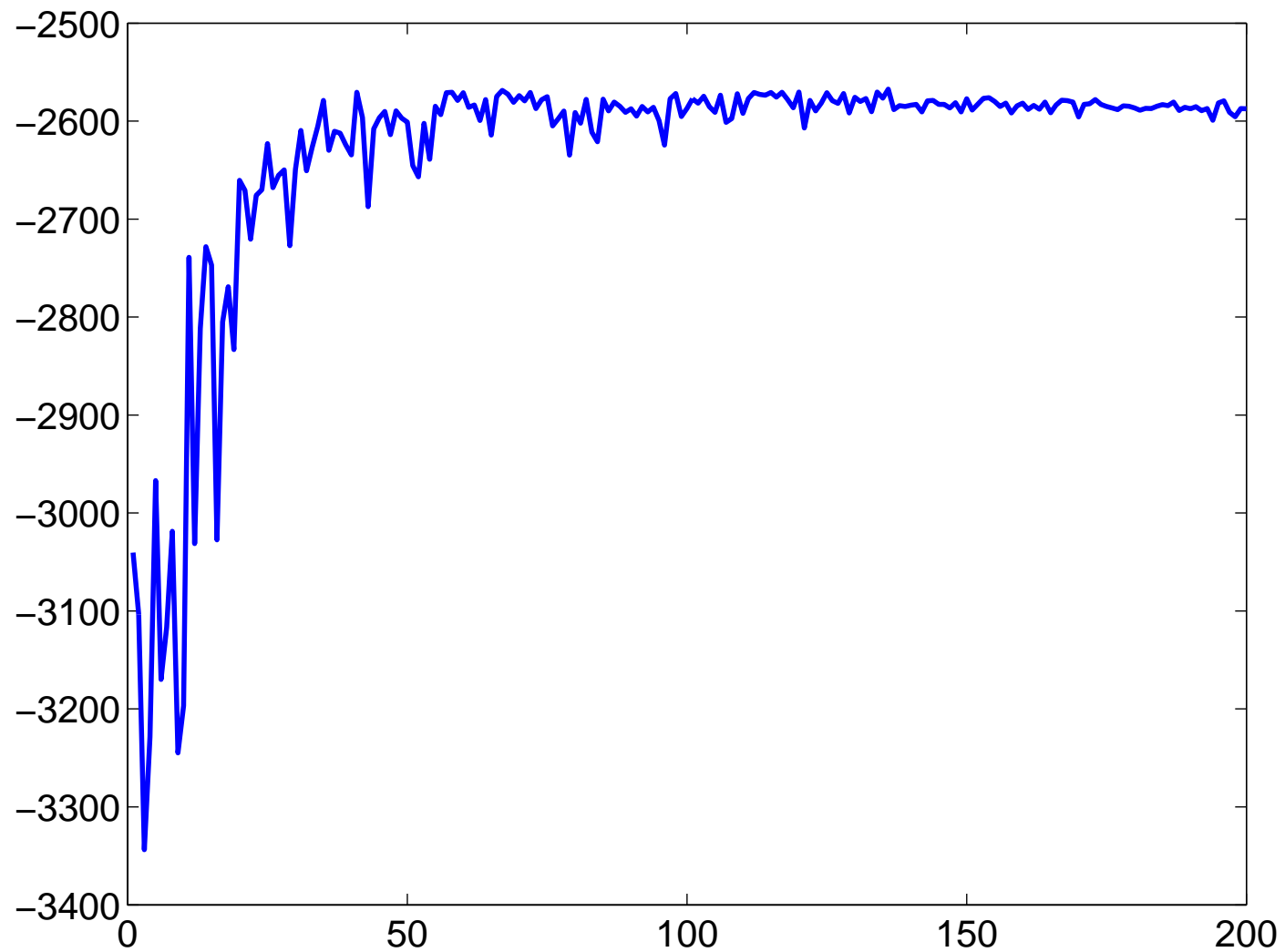
$$y' \in [0, m'] \quad \text{with probability} \quad P(y') = \frac{1}{m'}$$
$$x' = \frac{m'}{m}x \quad \text{with probability} \quad P(x') = P(x) \frac{dx}{dx'} = P(x) \frac{m}{m'}$$

Backward move

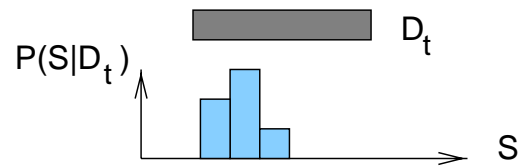
$$y \in [0, m] \quad \text{with probability} \quad P(y) = \frac{1}{m}$$
$$x \quad \text{with probability} \quad P(x)$$

Hastings ratio

$$\frac{P(\text{backward move})}{P(\text{forward move})} = \frac{P(y)P(x)}{P(y')P(x')} = \left(\frac{m'}{m}\right)^2$$



Marginal posterior distribution of tree topologies with MCMC



$$P(S|\mathcal{D}_t) = \int P(S, \mathbf{w}|\mathcal{D}_t) d\mathbf{w}$$

MCMC \longrightarrow Sample : $\{S_{ti}, \mathbf{w}_{ti}\}_{i=1}^N$

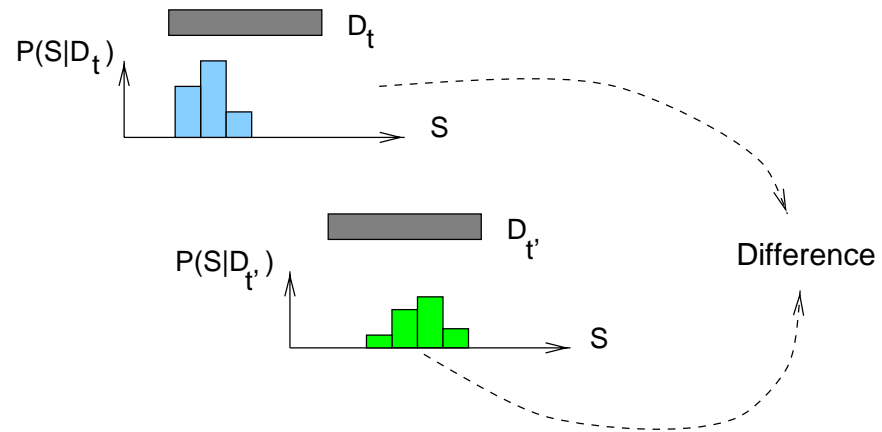
$$P(S, \mathbf{w}|\mathcal{D}_t) \approx \frac{1}{N} \sum_{i=1}^N \delta_{S, S_{ti}} \delta(\mathbf{w} - \mathbf{w}_{ti})$$

$$P(S|\mathcal{D}_t) = \frac{1}{N} \sum_{i=1}^N \delta_{S, S_{ti}} = \frac{N_S(t)}{N}$$

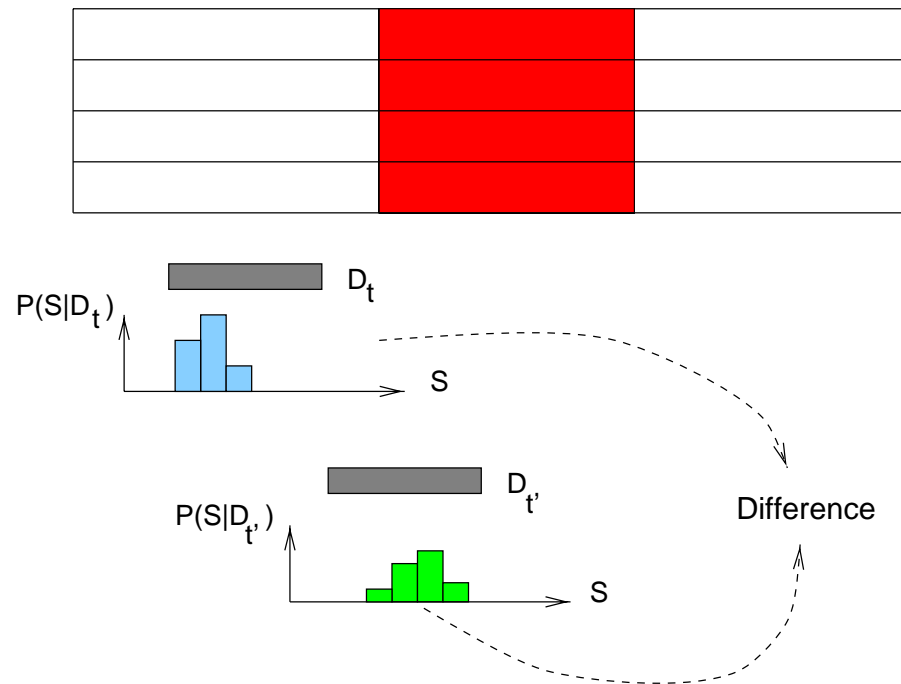
PDM method

- Posterior probability distribution over tree topologies
- Probabilistic divergence measure (PDM)
- Significance estimation

Divergence between distributions



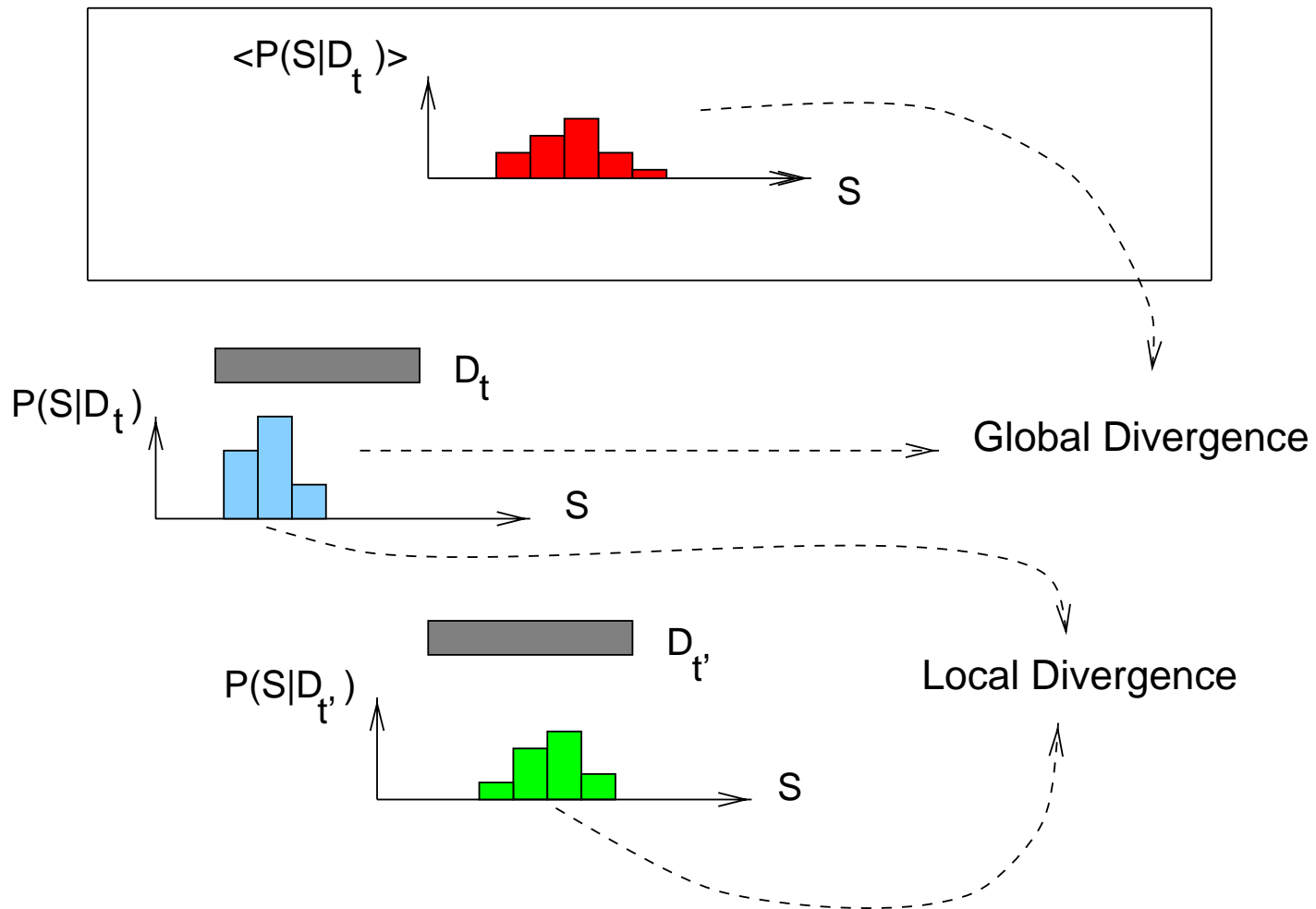
Divergence between distributions



Divergence measure in probability space: **Kullback-Leibler divergence**

$$KL(P, Q) = \sum_S P_S \ln \left(\frac{P_S}{Q_S} \right)$$

Local and global divergence measures



Local and global divergence measures

Global divergence between the distribution over the window, $P_S(t)$, and the average distribution, $\bar{P} = \frac{1}{W} \sum_{t=1}^W P_S(t)$:

$$d[P_S(t), \bar{P}] = \sum_S P_S(t) \ln \left(\frac{P_S(t)}{\bar{P}_S} \right)$$

Local and global divergence measures

Global divergence between the distribution over the window, $P_S(t)$, and the average distribution, $\bar{P} = \frac{1}{W} \sum_{t=1}^W P_S(t)$:

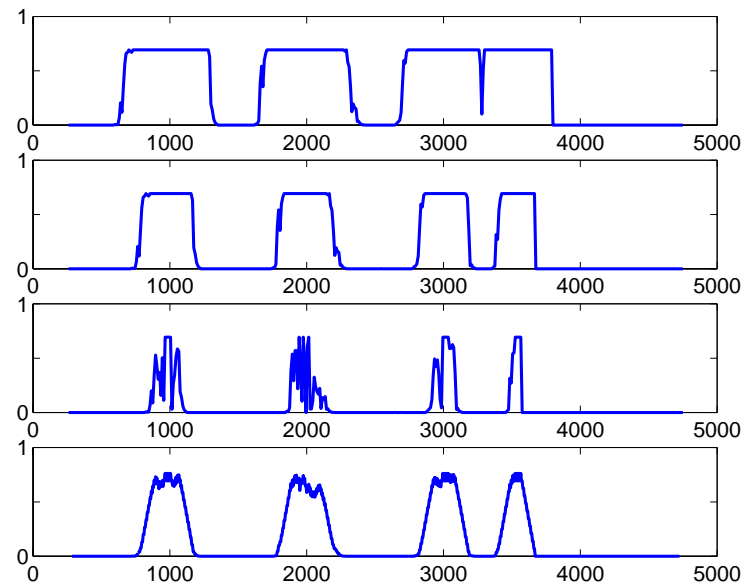
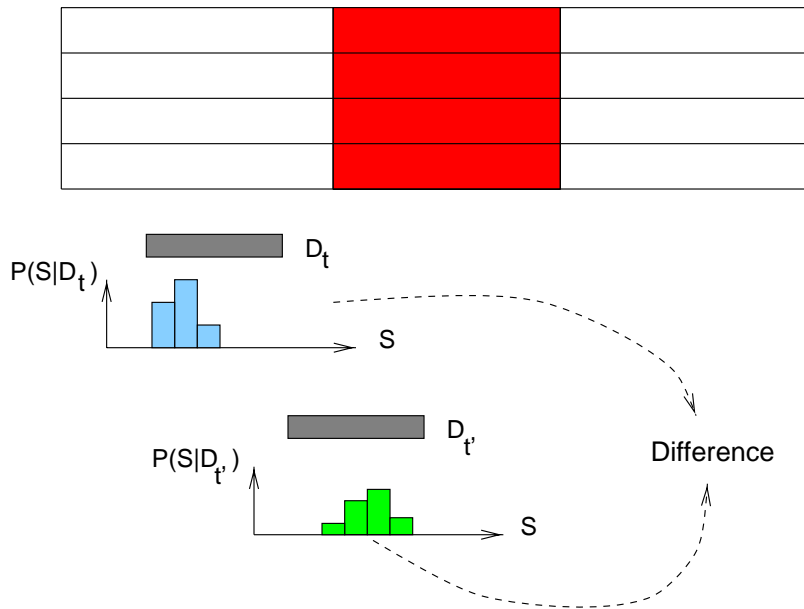
$$d[P_S(t), \bar{P}] = \sum_S P_S(t) \ln \left(\frac{P_S(t)}{\bar{P}_S} \right)$$

Local divergence between the distributions over two adjacent windows, $P_S(t)$ and $P_S(t')$, where $\tilde{P}_S = \frac{P_S(t) + P_S(t')}{2}$ (Sibson):

$$d[P_S(t), P_S(t')] = \frac{1}{2} \sum_S \left[P_S(t) \ln \left(\frac{P_S(t)}{\tilde{P}_S} \right) + P_S(t') \ln \left(\frac{P_S(t')}{\tilde{P}_S} \right) \right]$$

Local Divergence Measure: Window Overlap

True breakpoints at positions 1000, 2000, 3000, 3500
Window size: 500



Left, top to bottom: 0%, 50%, 90% overlap, averaging

PDM method

- Posterior probability distribution over tree topologies
- Probabilistic divergence measure (PDM)
- Significance estimation

Asymptotic analytic results (sample size $M \rightarrow \infty$)

Divergence between the distribution over the window, $P_S(t)$, and the average distribution, $\bar{P} = \frac{1}{W} \sum_{t=1}^W P_S(t)$:

$$d[P_S(t), \bar{P}] = \sum_S P_S(t) \ln \left(\frac{P_S(t)}{\bar{P}_S} \right)$$

Divergence between the distributions over two adjacent windows, $P_S(t)$ and $P_S(t')$, where $\tilde{P}_S = \frac{P_S(t) + P_S(t')}{2}$ (Sibson):

$$d[P_S(t), P_S(t')] = \frac{1}{2} \sum_S \left[P_S(t) \ln \left(\frac{P_S(t)}{\tilde{P}_S} \right) + P_S(t') \ln \left(\frac{P_S(t')}{\tilde{P}_S} \right) \right]$$

Null hypotheses: $P_S(t) = \bar{P}_S$ and $P_S(t) = P_S(t')$

$$\begin{aligned} 2Md[P_S(t), \bar{P}] &\rightarrow \chi^2(\nu - 1), & \nu &= |\text{Support}(\bar{P})| \\ 2Md[P_S(t), P_S(t')] &\rightarrow \chi^2(\tilde{\nu} - 1), & \tilde{\nu} &= |\text{Support}(\tilde{P})| \end{aligned}$$

Avoid asymptotics

Bootstrapping

Generate sequence alignments under the null hypothesis of no recombination

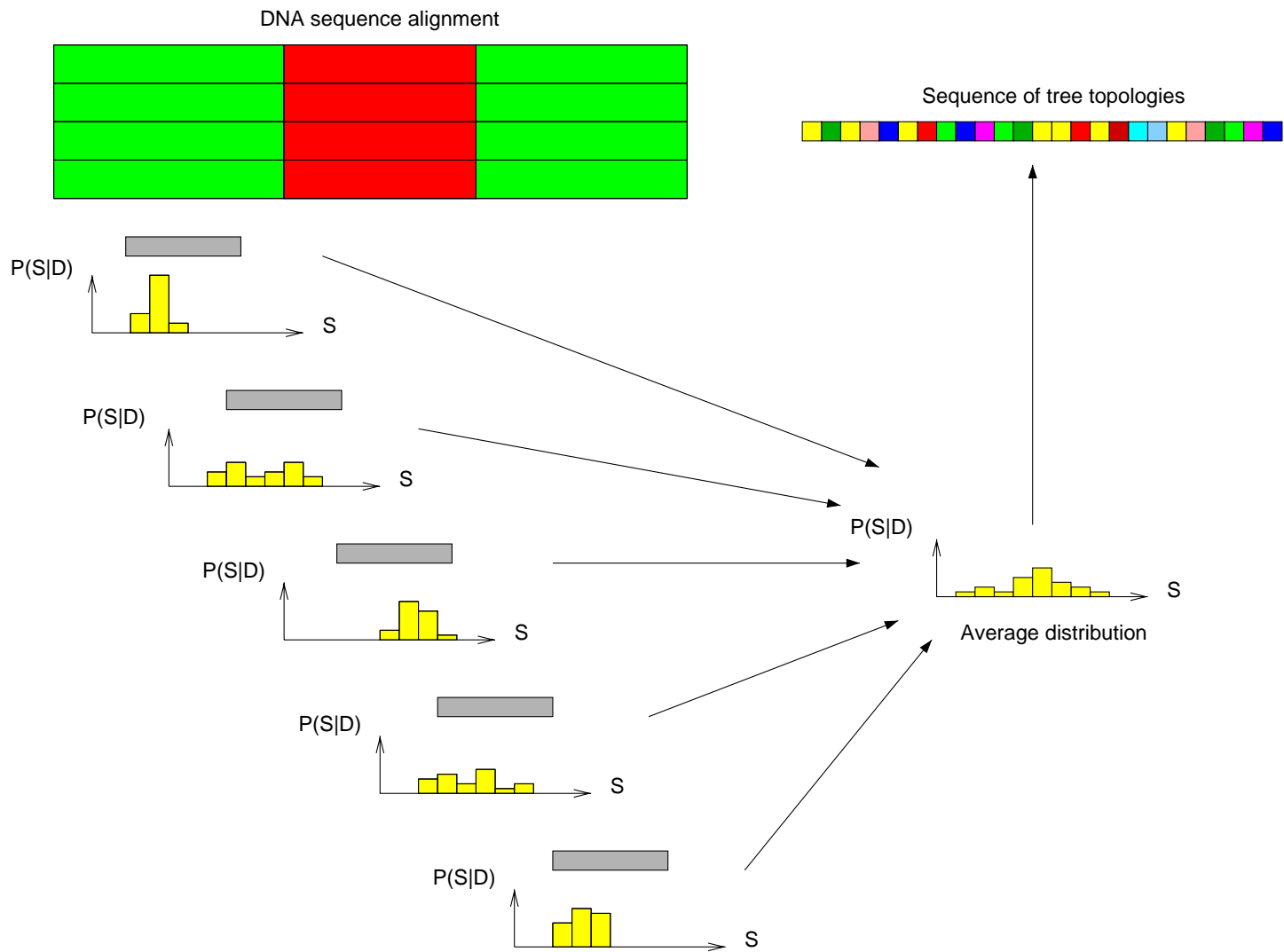
Repeat the analysis

→ Distribution of peak heights under the null hypothesis

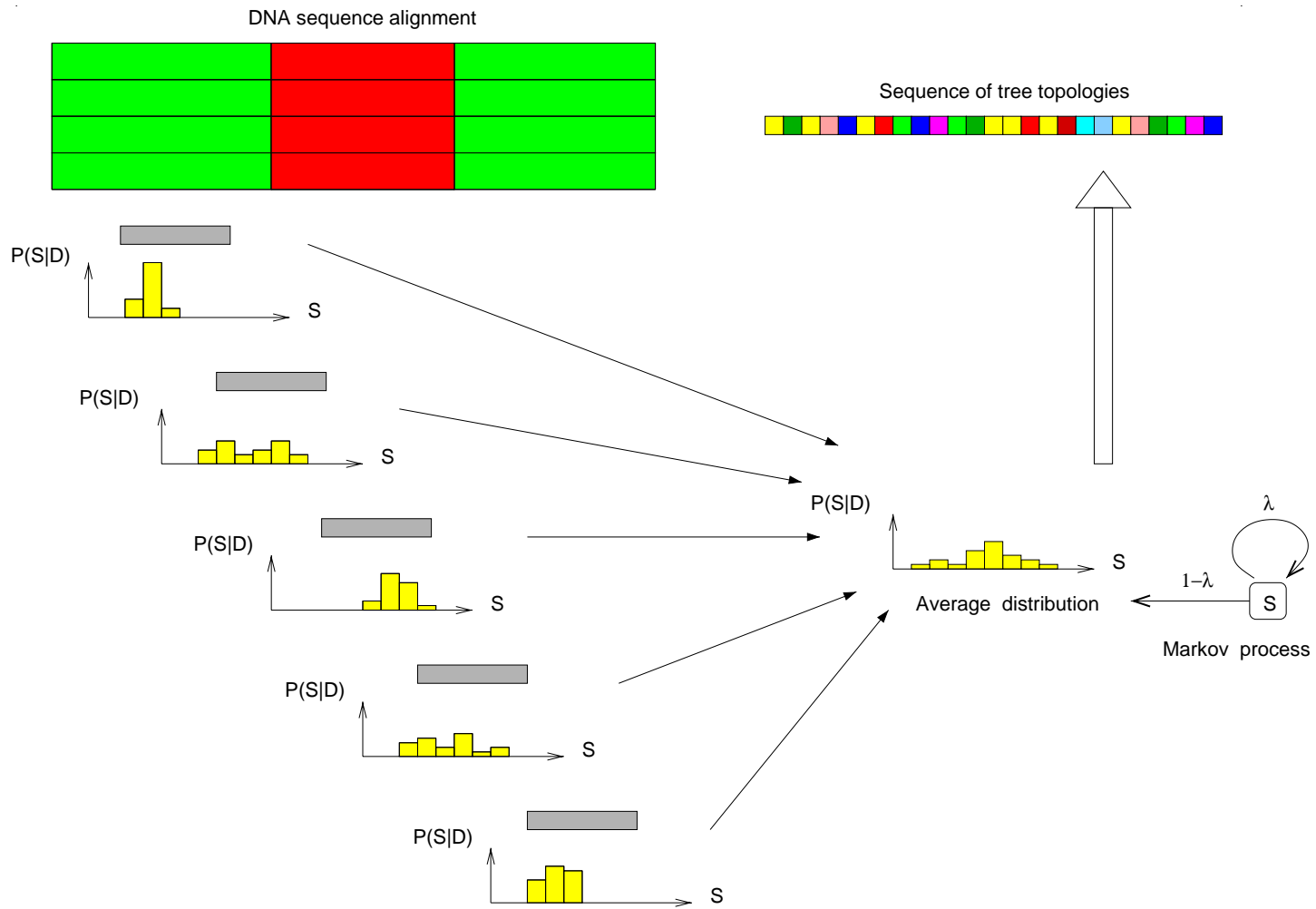
Shortcoming

Computational costs exorbitant

Sort of "parametric bootstrapping"



Sort of "parametric bootstrapping"



Software implementation

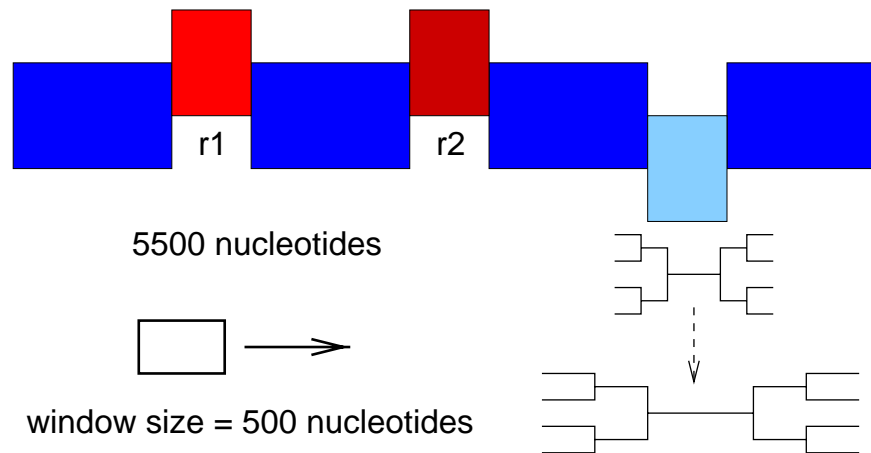
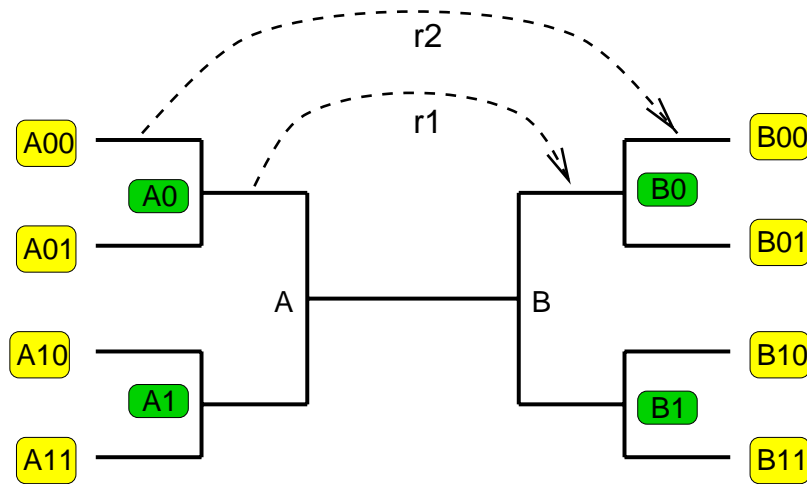
Milne, Wright, Rowe, Marshall, Husmeier, McGuire

TOPALi: Software for automatic identification of recombinant sequences within DNA multiple alignments

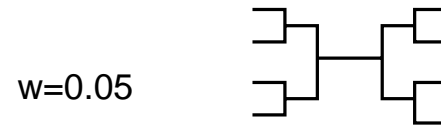
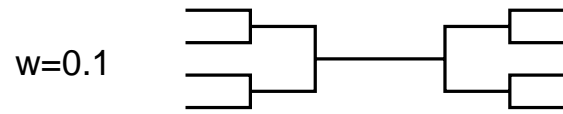
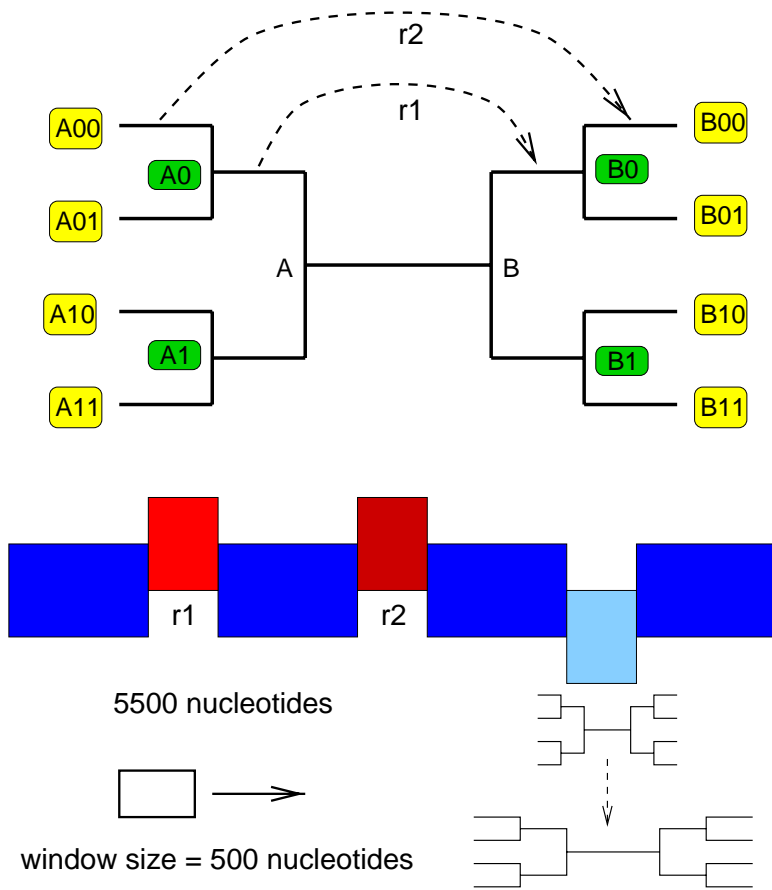
Bioinformatics 2004, in press

Application and evaluation

Simulation experiment

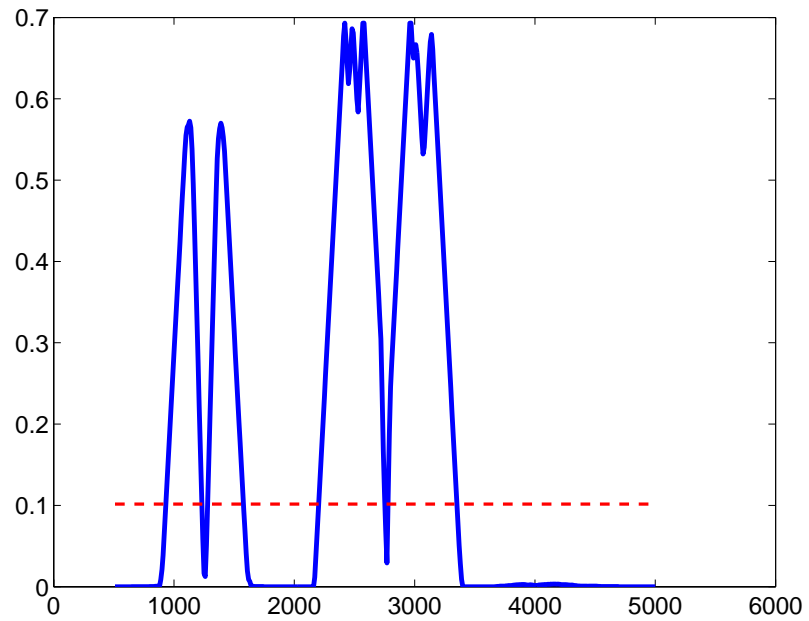
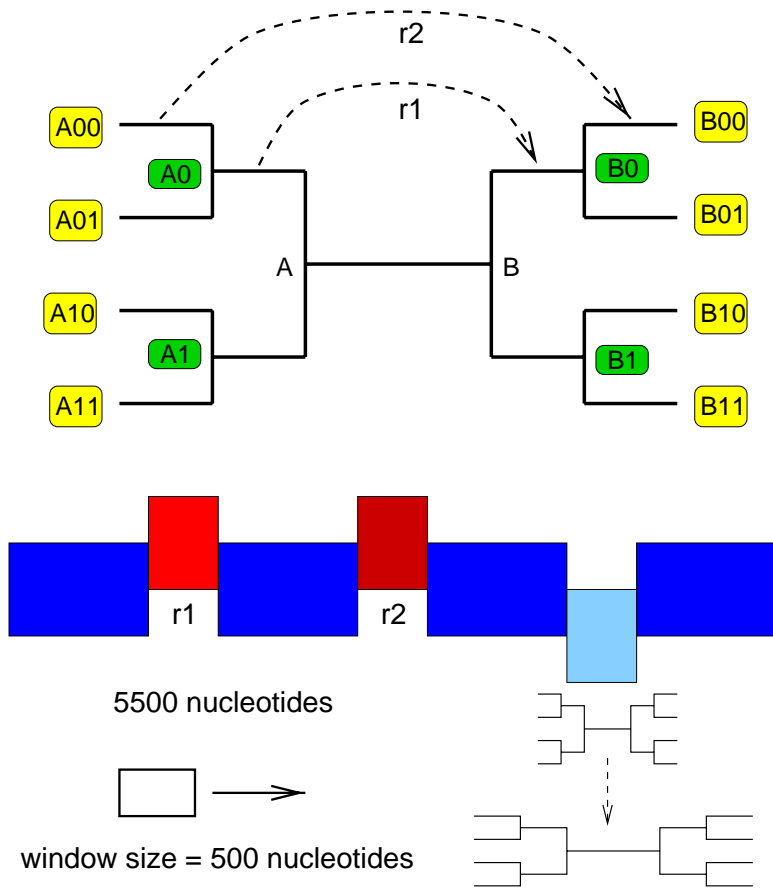


Varying branch lengths: $w = 0.1 \rightarrow w = 0.01$

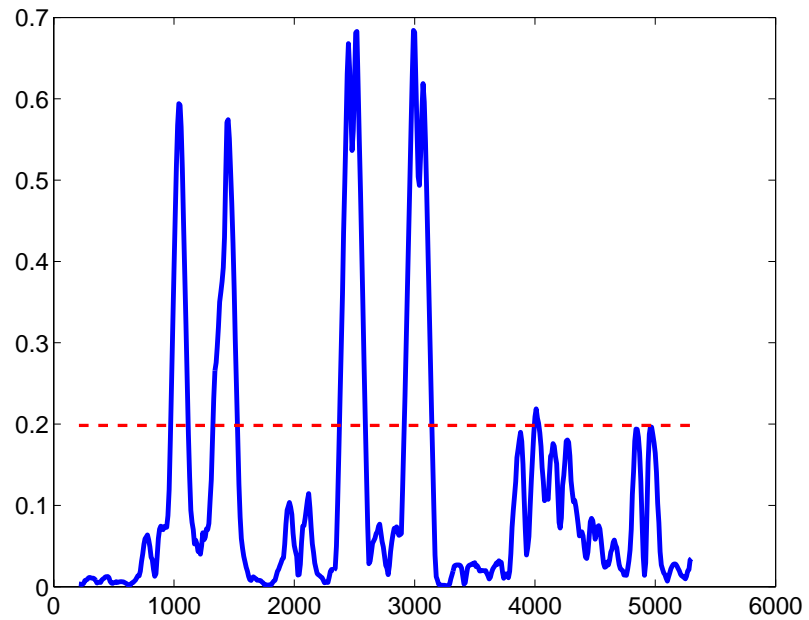
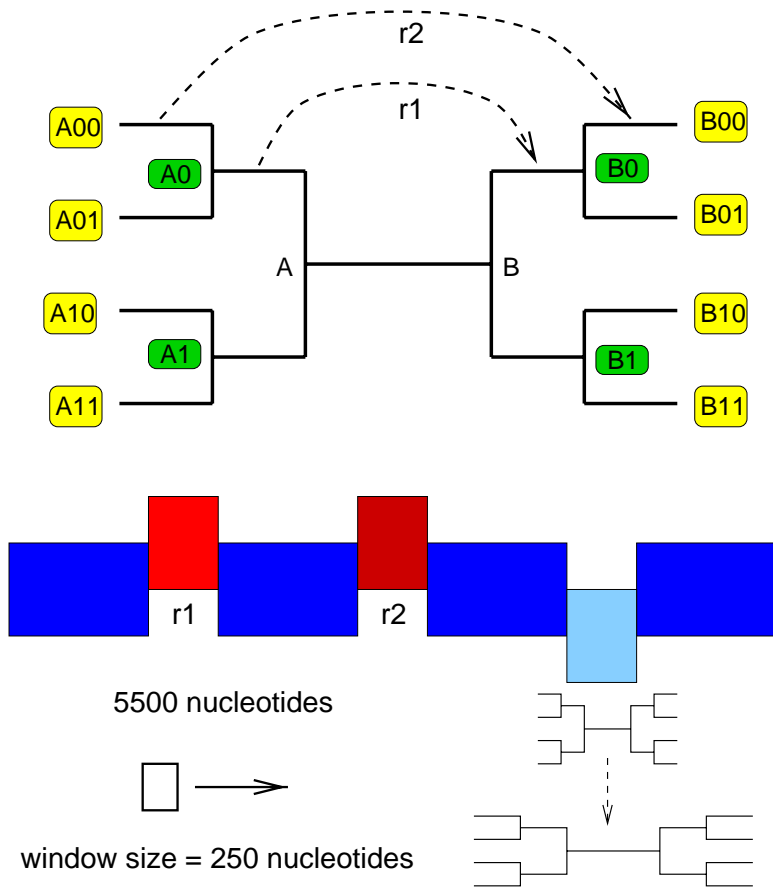


Decreasing
sequence
divergence

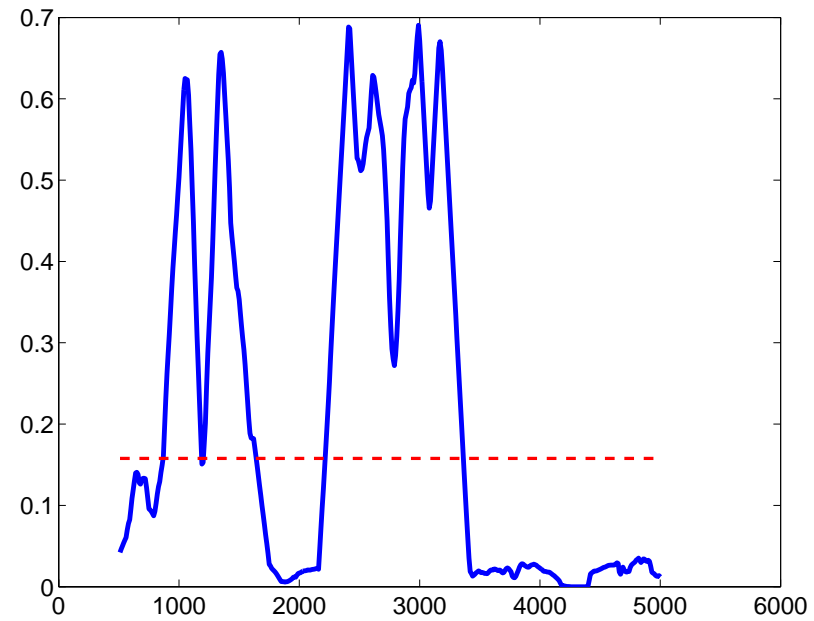
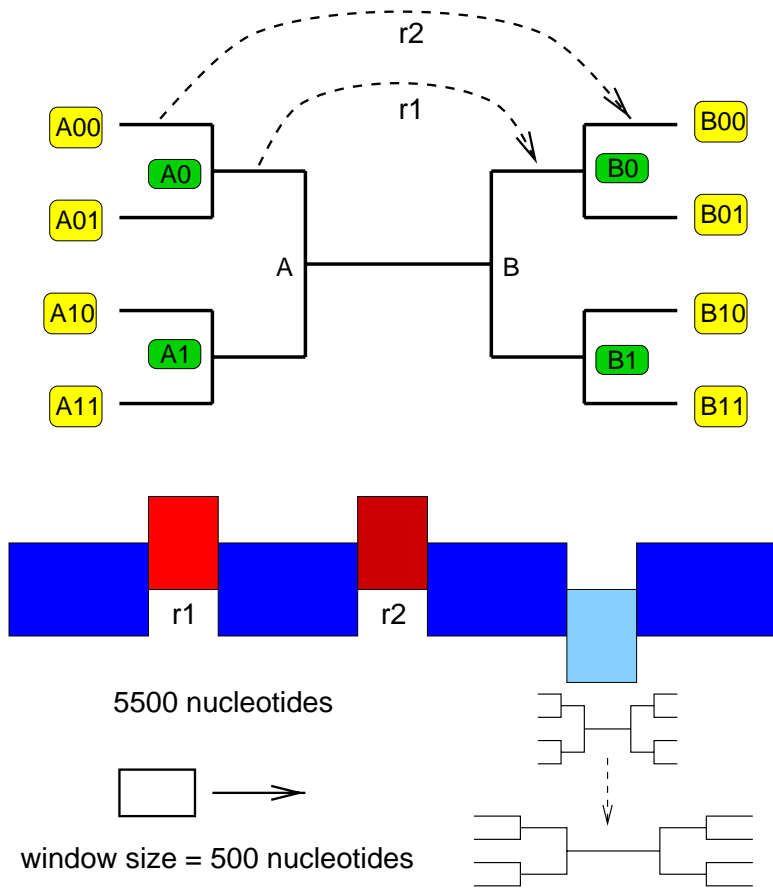
Sequence divergence: **HIGH** ($w = 0.1$). Window = 500 bp



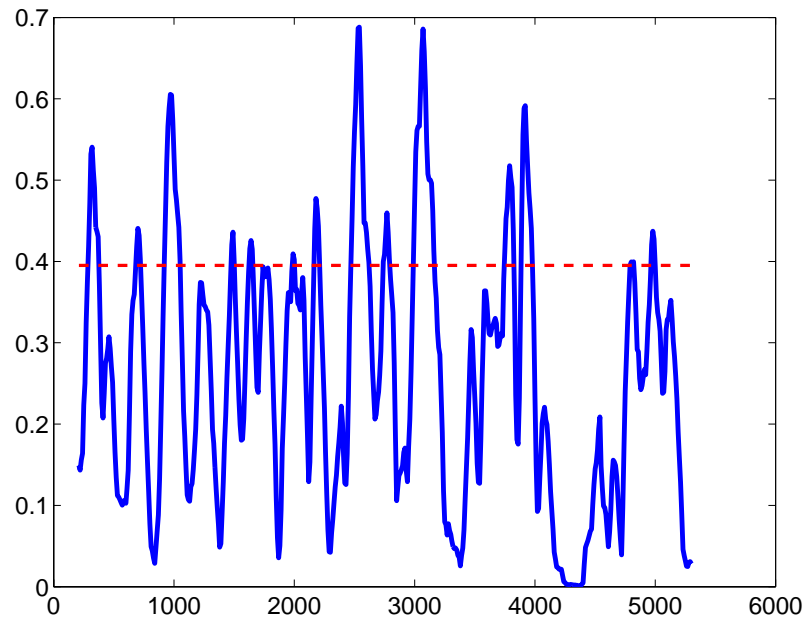
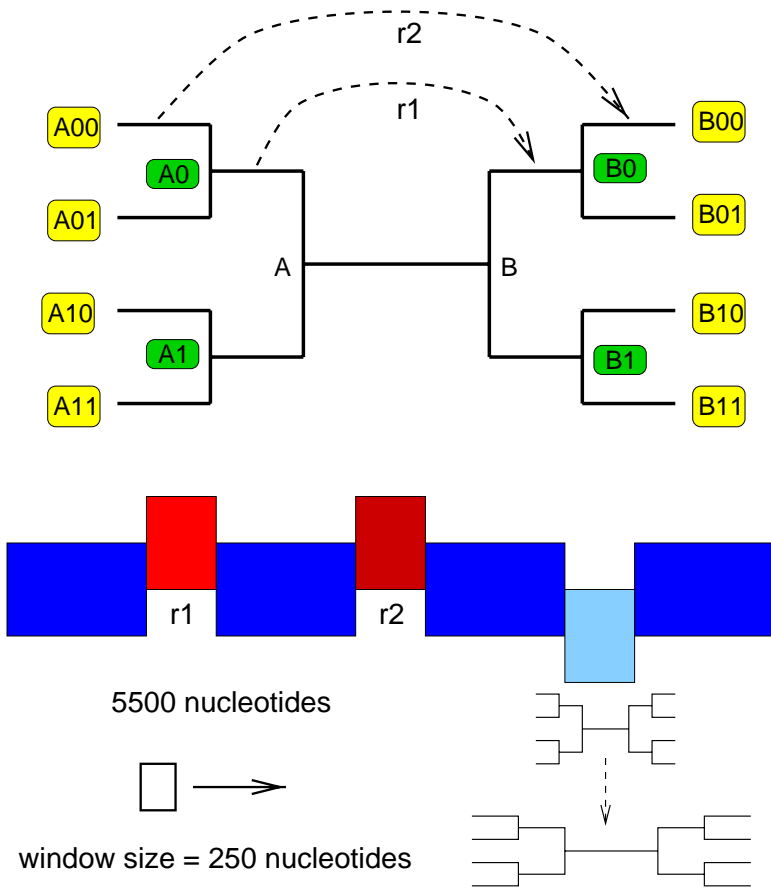
Sequence divergence: **HIGH** ($w = 0.1$). Window = 200 bp



Sequence divergence: **LOW** ($w = 0.01$). Window = 500 bp



Sequence divergence: **LOW** ($w = 0.01$). Window = 200 bp

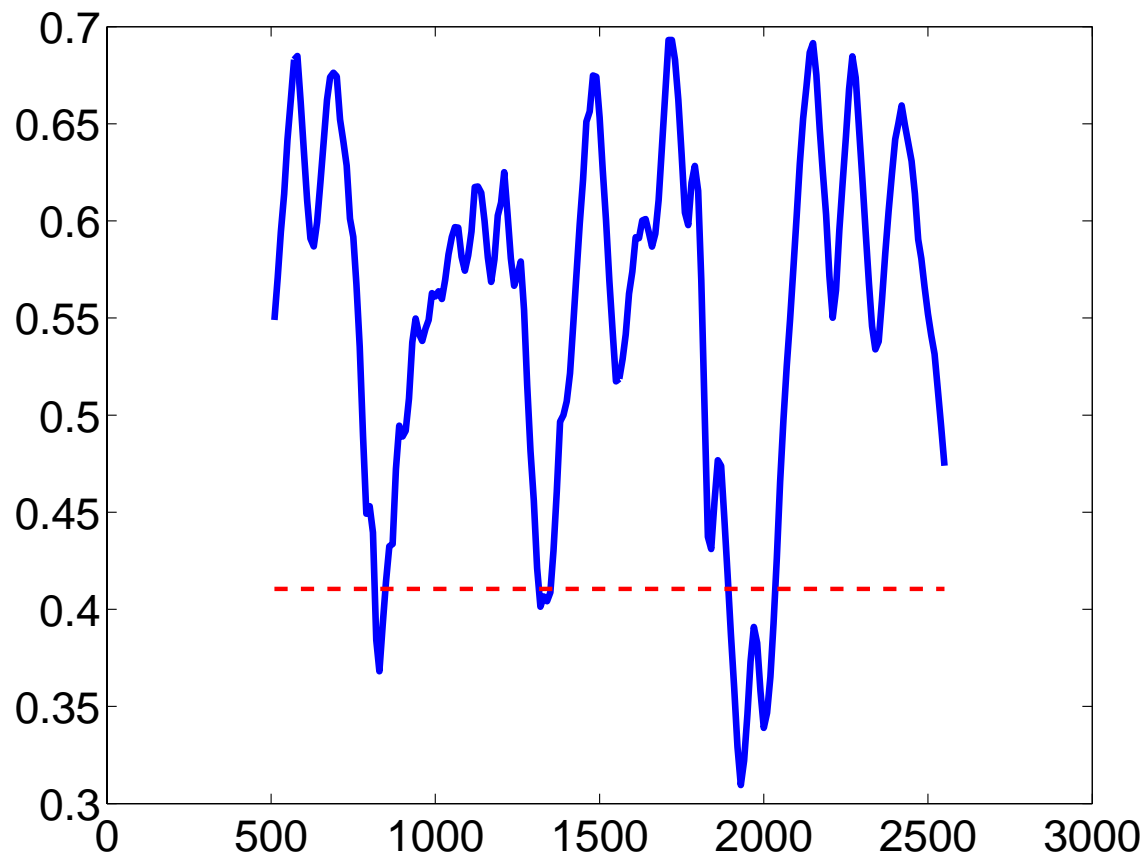


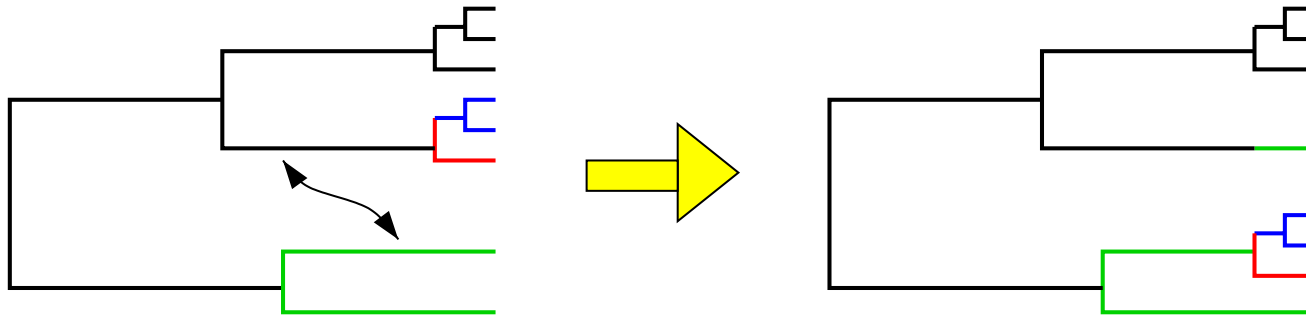
Detecting recombination with window methods

- **DSS (TOPAL)**
McGuire, Wright, Prentice (Mol. Biol. Evol. 14)
- **PDM**
Husmeier, Wright (Bioinformatics 18)
- **Pruned PDM**
Work in progress

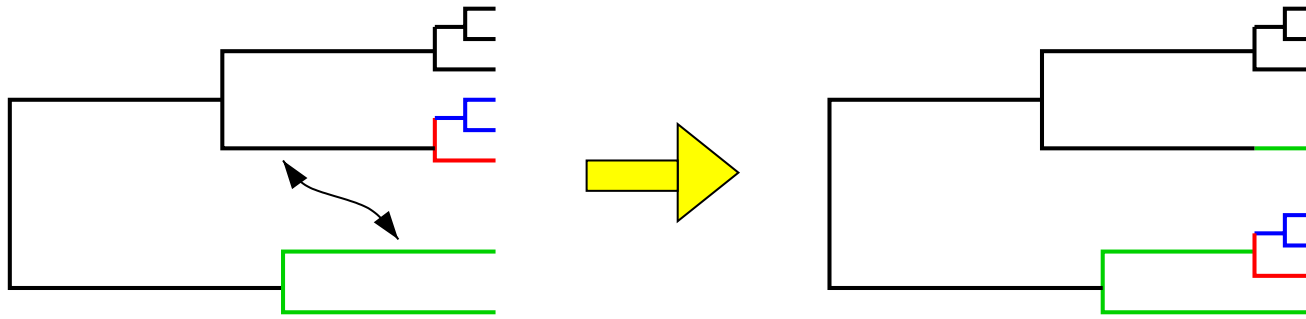
Problem of PDM

10 strains of Hepatitis B virus, DNA sequence alignment: 3049 bp
→ 126 distinct topologies

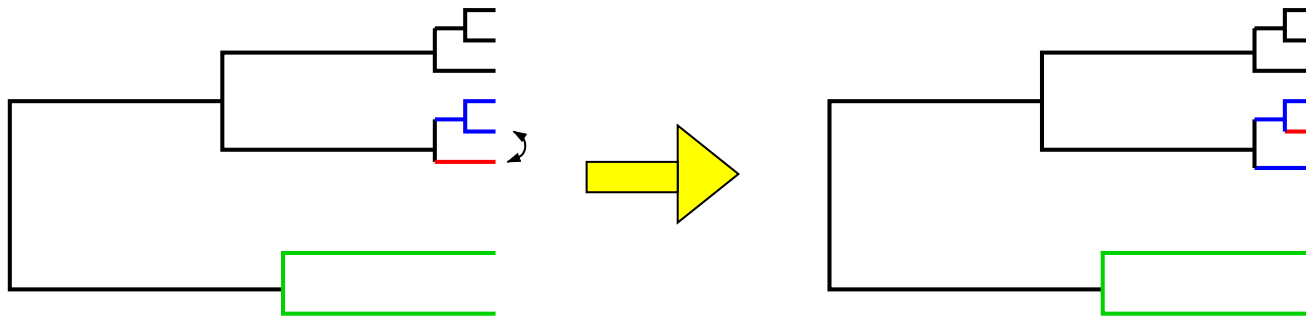




Substantial change



Substantial change



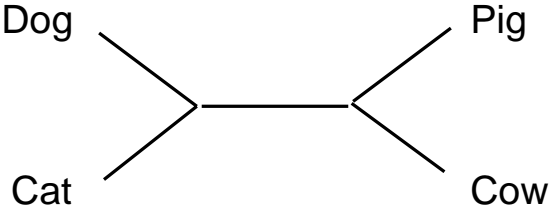
Small change

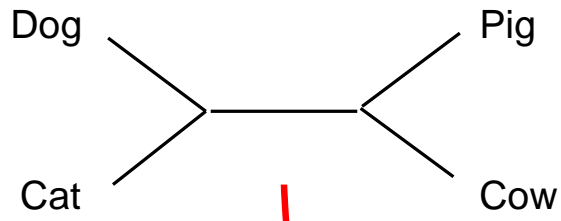
How can we measure the distance between tree topologies?

How can we measure the distance between tree topologies?

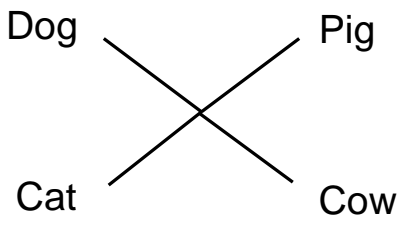
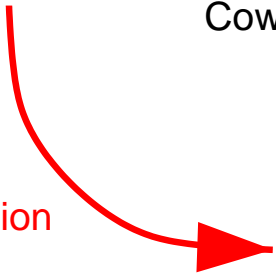
Robinson-Foulds distance

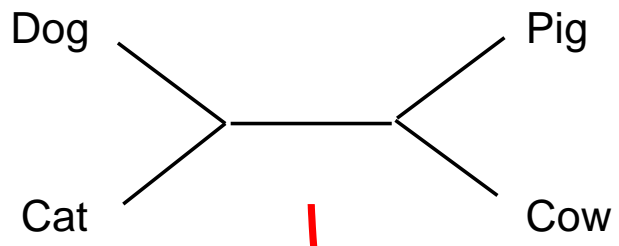
Robinson, D.R. and Foulds, L.R. (1981)
Mathematical Biosciences 53, 131-147



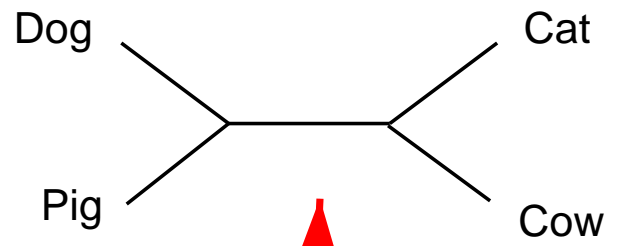
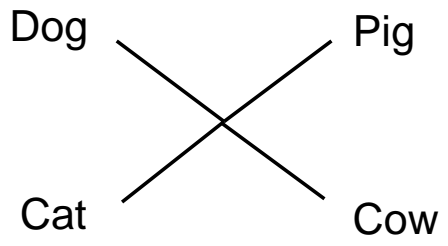
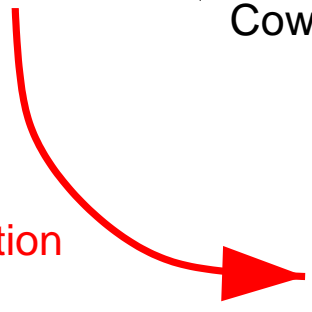


Contraction

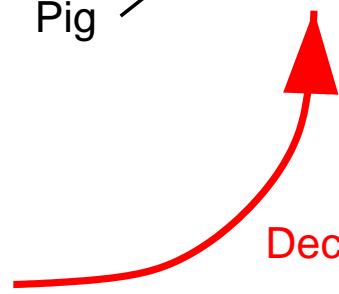




Contraction



Decontraction



Robinson-Foulds distance

- Edge \longrightarrow Bi-partition of taxa (leaves)

Robinson-Foulds distance

- **Edge** \longrightarrow Bi-partition of taxa (leaves)
- **E(Tree)** : Set of all bi-partitions

Robinson-Foulds distance

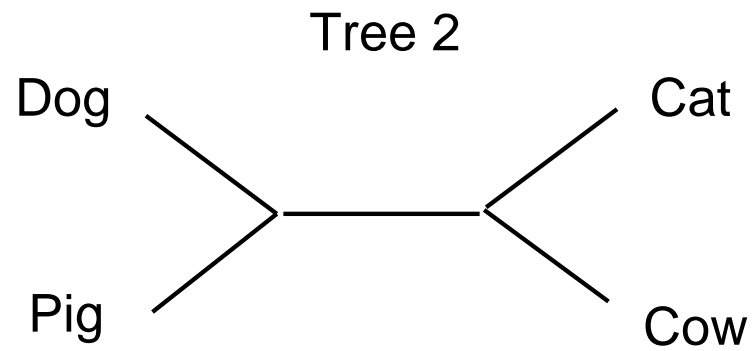
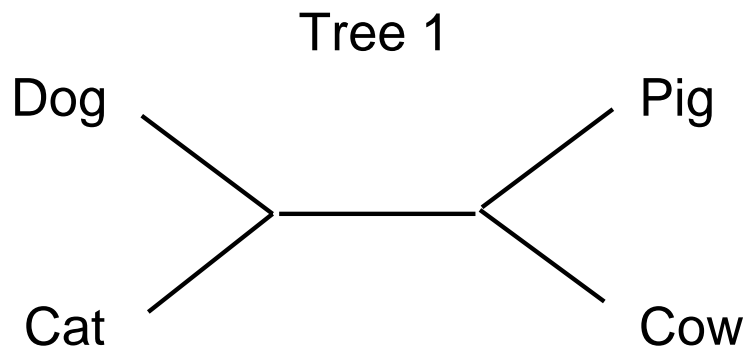
- Edge \longrightarrow Bi-partition of taxa (leaves)
- $E(\text{Tree})$: Set of all bi-partitions
- $|E(\text{Tree1})-E(\text{Tree2})|$: Number of bipartitions induced by edges in Tree1 that cannot be found in Tree2.

Robinson-Foulds distance

- **Edge** \longrightarrow Bi-partition of taxa (leaves)
- **$E(\text{Tree})$** : Set of all bi-partitions
- **$|E(\text{Tree1})-E(\text{Tree2})|$** : Number of bipartitions induced by edges in Tree1 that cannot be found in Tree2.
- **$|E(\text{Tree2})-E(\text{Tree1})|$** : Number of bipartitions induced by edges in Tree2 that cannot be found in Tree1.

Robinson-Foulds distance

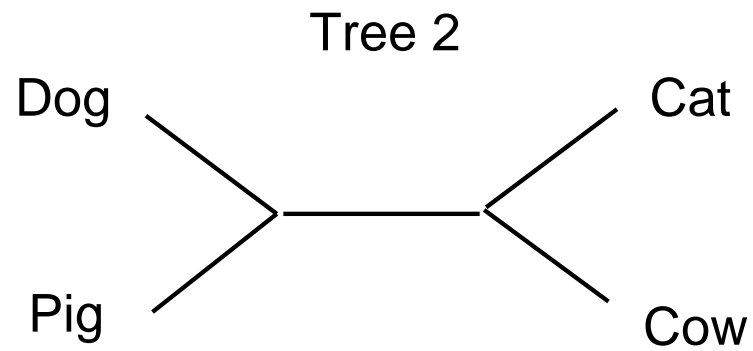
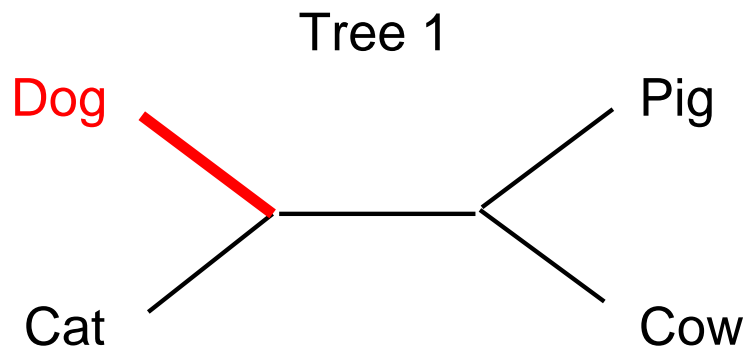
- **Edge** \longrightarrow Bi-partition of taxa (leaves)
- **$E(\text{Tree})$** : Set of all bi-partitions
- **$|E(\text{Tree1})-E(\text{Tree2})|$** : Number of bipartitions induced by edges in Tree1 that cannot be found in Tree2.
- **$|E(\text{Tree2})-E(\text{Tree1})|$** : Number of bipartitions induced by edges in Tree2 that cannot be found in Tree1.
- **RF-distance** = $|E(\text{Tree1})-E(\text{Tree2})| + |E(\text{Tree2})-E(\text{Tree1})|$



$$|E(\text{Tree1})-E(\text{Tree2})| =$$

$$|E(\text{Tree2})-E(\text{Tree1})| =$$

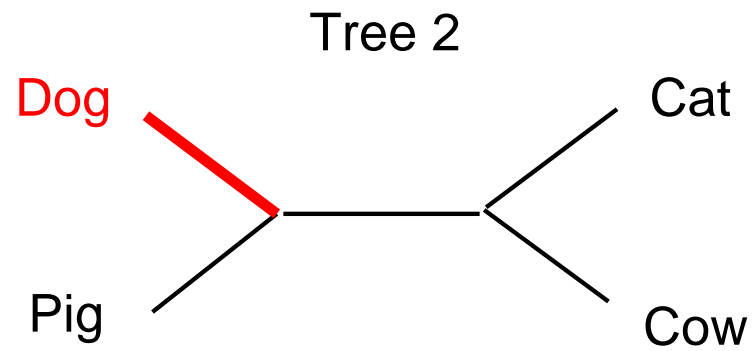
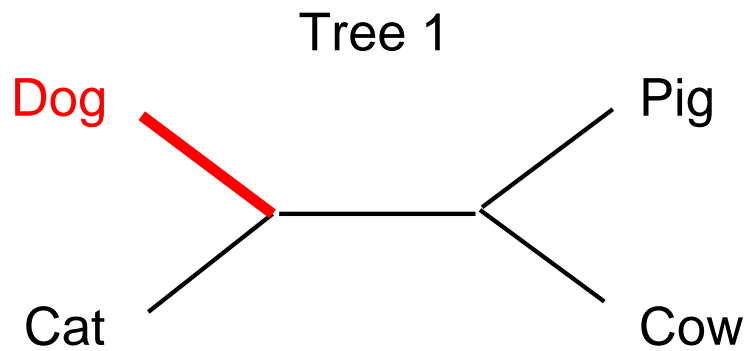
$$\text{RF distance} =$$



$|E(\text{Tree1})-E(\text{Tree2})| =$

$|E(\text{Tree2})-E(\text{Tree1})| =$

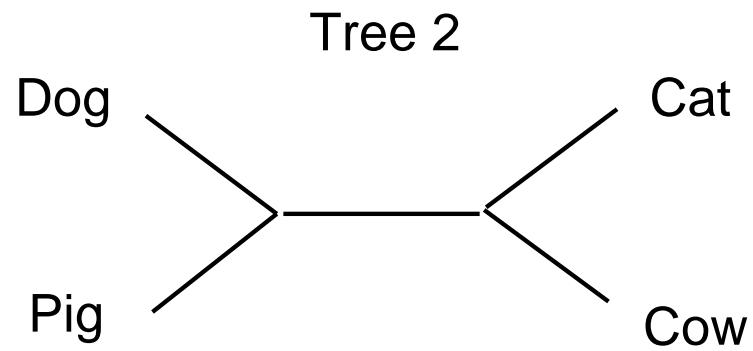
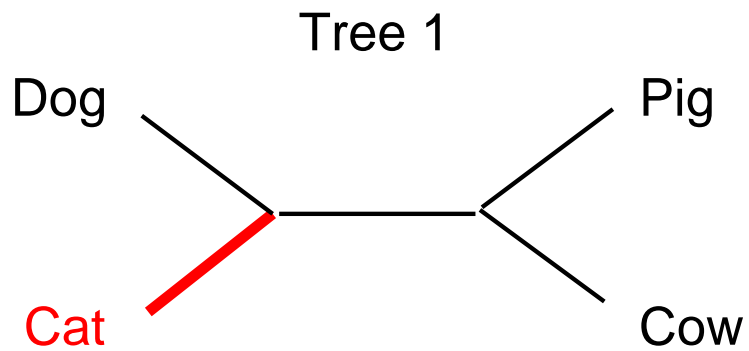
RF distance =



$$|E(\text{Tree1})-E(\text{Tree2})| =$$

$$|E(\text{Tree2})-E(\text{Tree1})| =$$

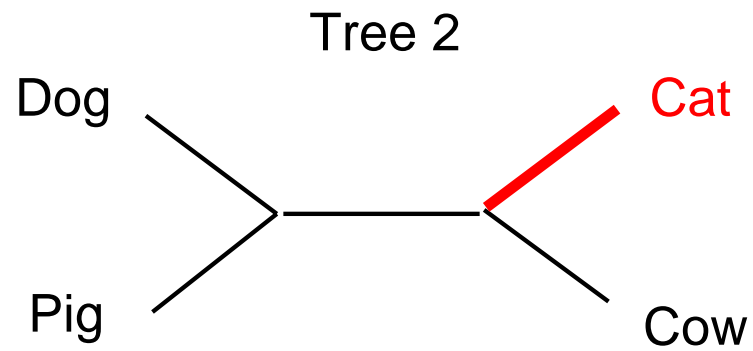
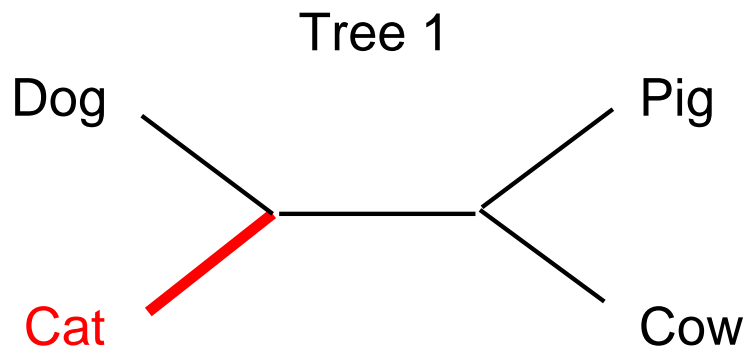
$$\text{RF distance} =$$



$$|E(\text{Tree1}) - E(\text{Tree2})| =$$

$$|E(\text{Tree2}) - E(\text{Tree1})| =$$

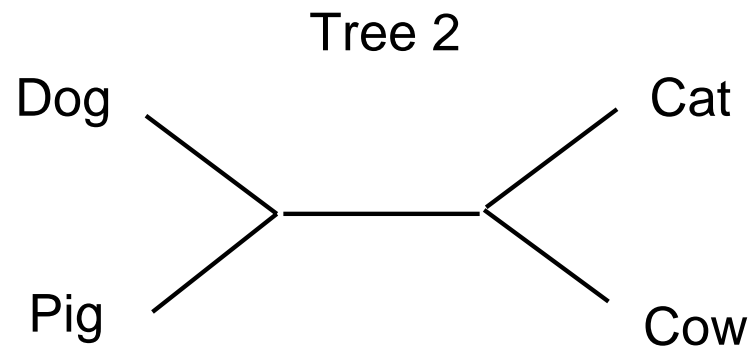
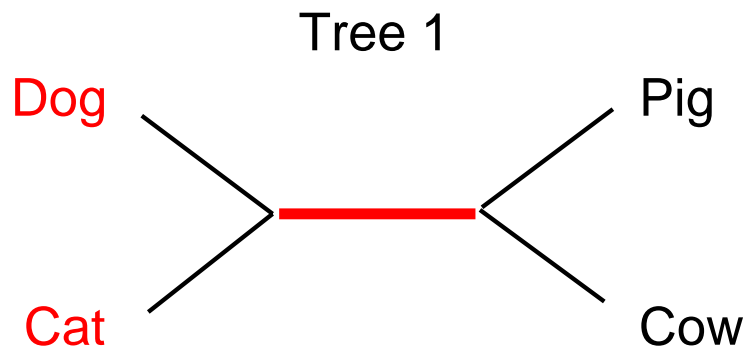
$$\text{RF distance} =$$



$|E(\text{Tree1})-E(\text{Tree2})| =$

$|E(\text{Tree2})-E(\text{Tree1})| =$

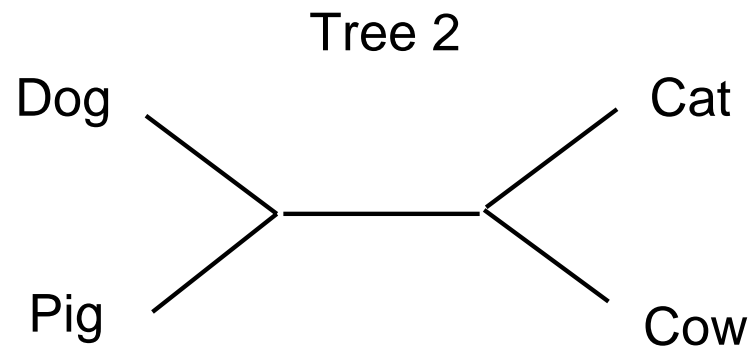
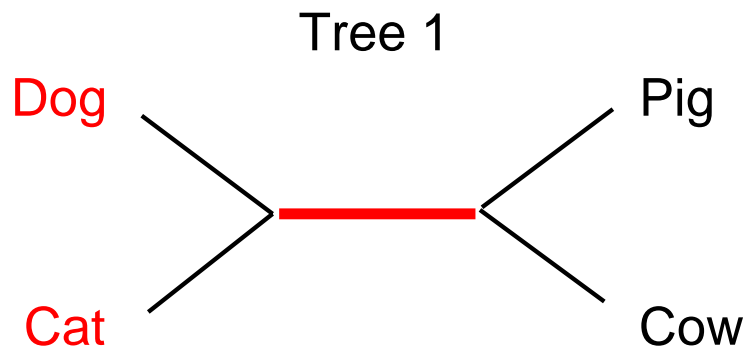
RF distance =



$$|E(\text{Tree1})-E(\text{Tree2})| =$$

$$|E(\text{Tree2})-E(\text{Tree1})| =$$

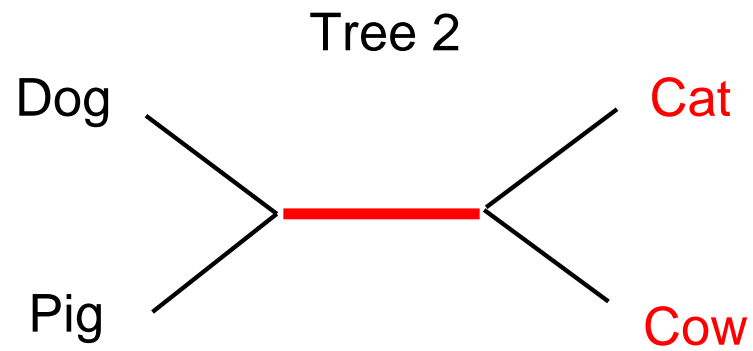
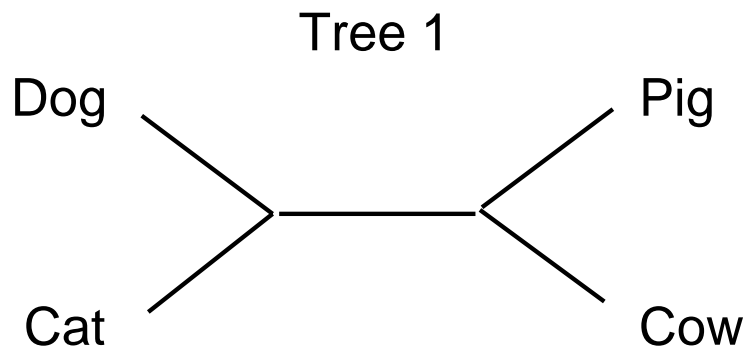
$$\text{RF distance} =$$



$$|E(\text{Tree1})-E(\text{Tree2})| = 1$$

$$|E(\text{Tree2})-E(\text{Tree1})| =$$

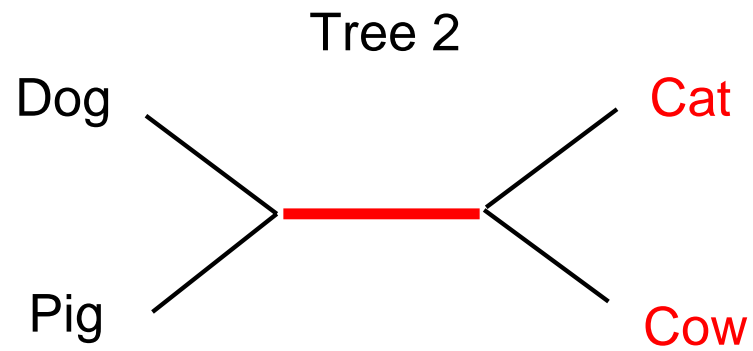
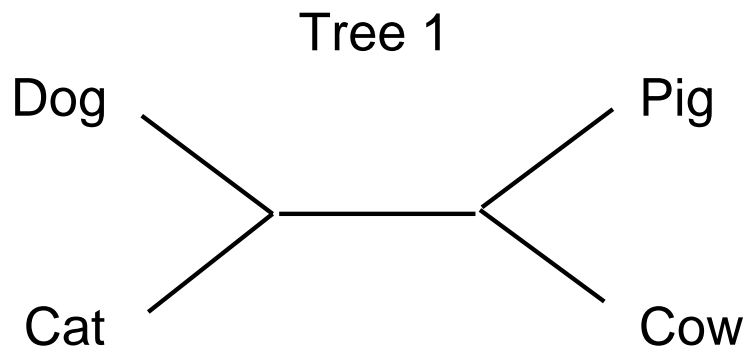
RF distance =



$$|E(\text{Tree1})-E(\text{Tree2})| = 1$$

$$|E(\text{Tree2})-E(\text{Tree1})| =$$

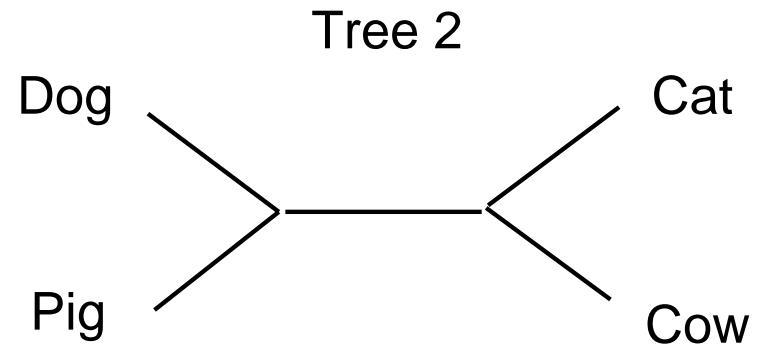
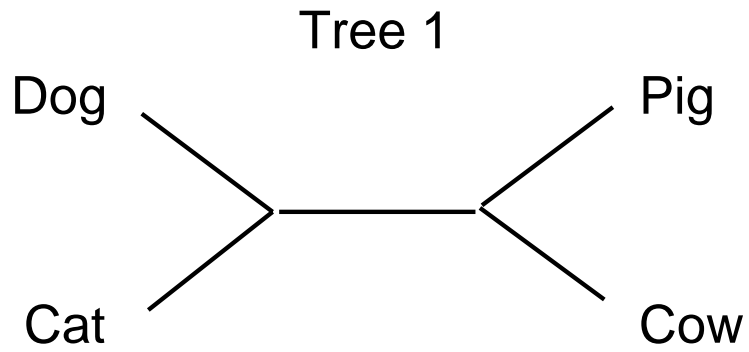
RF distance =



$$|E(\text{Tree1})-E(\text{Tree2})| = 1$$

$$|E(\text{Tree2})-E(\text{Tree1})| = 1$$

RF distance =

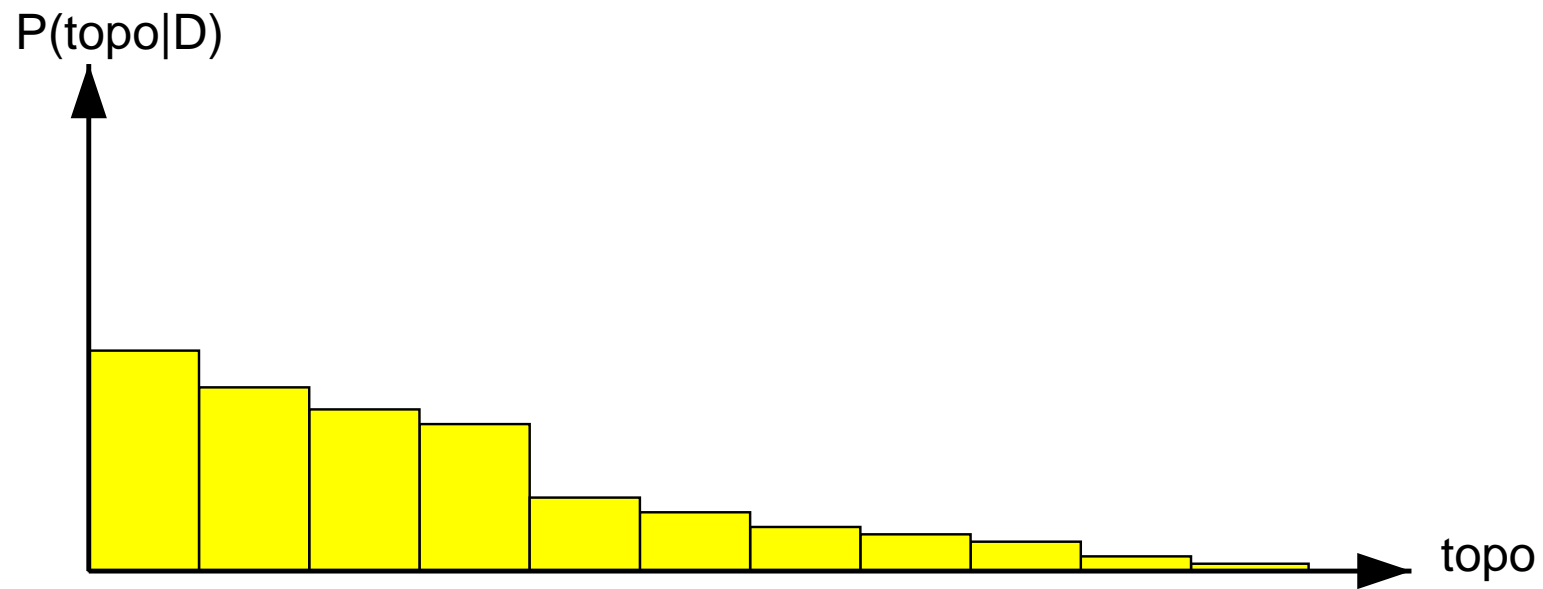


$$|E(\text{Tree1})-E(\text{Tree2})| = 1$$

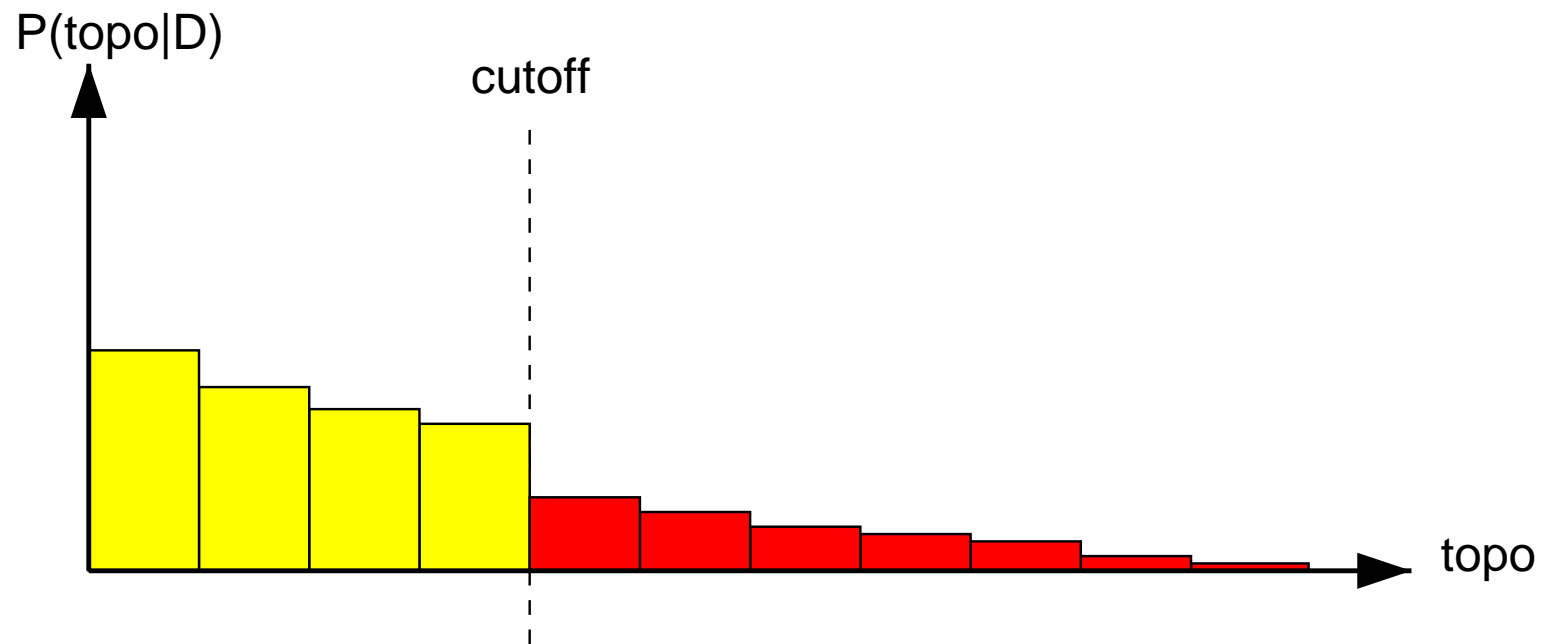
$$|E(\text{Tree2})-E(\text{Tree1})| = 1$$

$$\text{RF distance} = 2$$

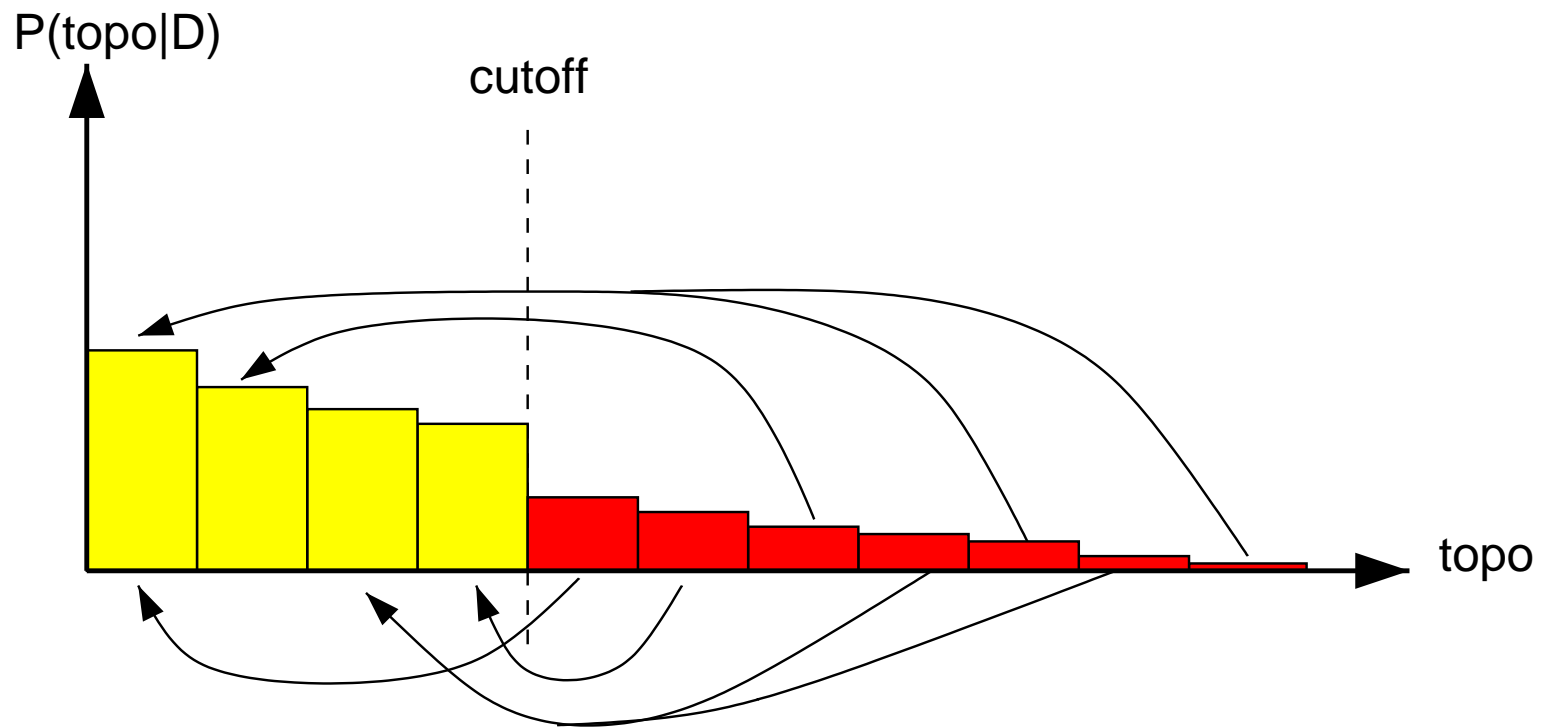
Pruning



Pruning

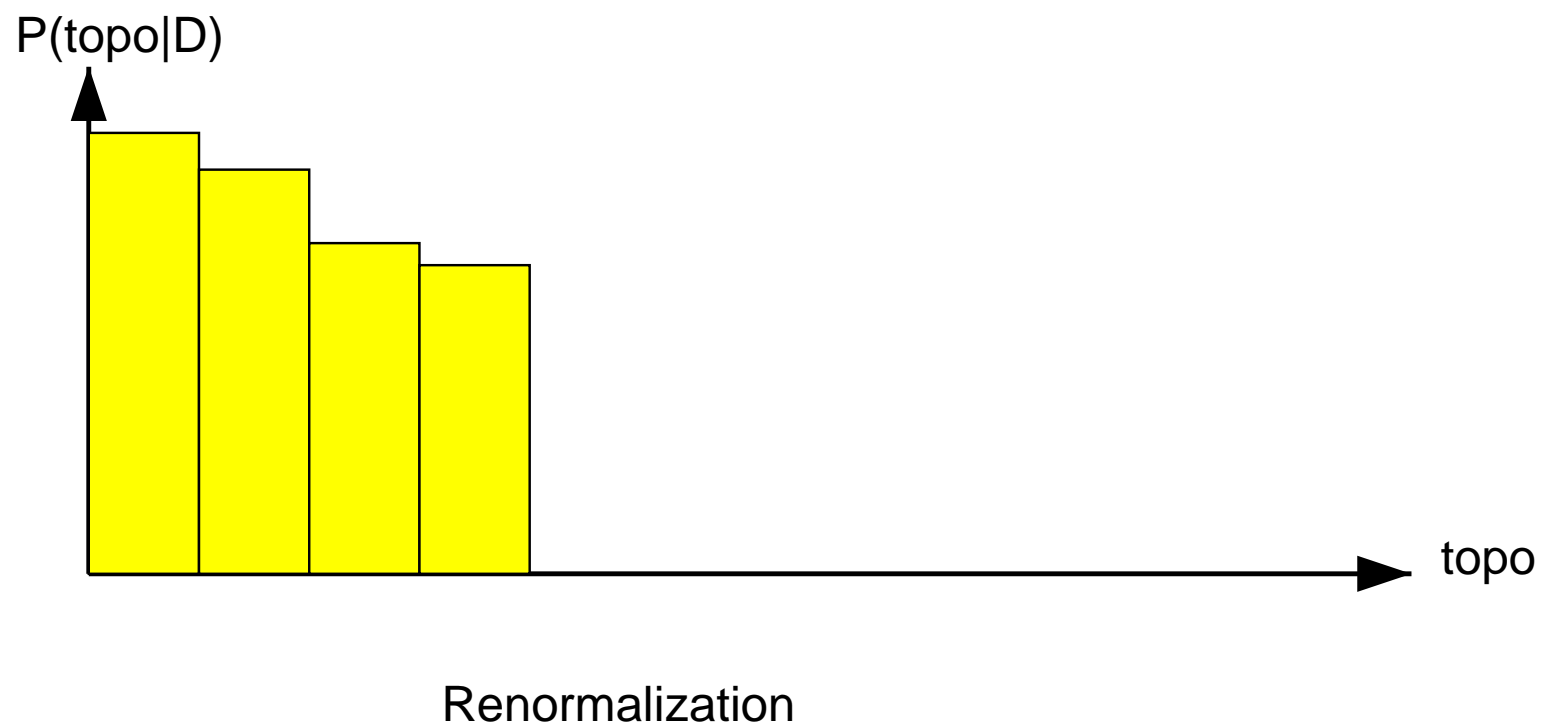


Pruning



Nearest neighbour assignment

Pruning



Pruning

- First step of **K-means** clustering.

Pruning

- First step of **K-means** clustering.
- Based on the **RF-distance**, cluster the complete set of tree topologies $\{S_t\}$ from MCMC simulations.

Pruning

- First step of **K-means** clustering.
- Based on the **RF-distance**, **cluster** the complete set of tree topologies $\{S_t\}$ from MCMC simulations.
- Assign tree topologies S_t to the **prototype** of their respective cluster, C .

Pruning

- First step of **K-means** clustering.
- Based on the **RF-distance**, **cluster** the complete set of tree topologies $\{S_t\}$ from MCMC simulations.
- Assign tree topologies S_t to the **prototype** of their respective cluster, C .
- Compute the distribution over prototypes:
 $P(S|D) \rightarrow P(C|D)$

Pruning

- First step of **K-means** clustering.
- Based on the **RF-distance**, cluster the complete set of tree topologies $\{S_t\}$ from MCMC simulations.
- Assign tree topologies S_t to the **prototype** of their respective cluster, C .
- Compute the distribution over prototypes:
 $P(S|D) \rightarrow P(C|D)$
- Compute the **(pruned) PDM signal**

Related work

Stockham, Wang, Warnow (2002)

Statistically based postprocessing of phylogenetic analysis by clustering

ISMB 2002

Bioinformatics 18, S285–S293

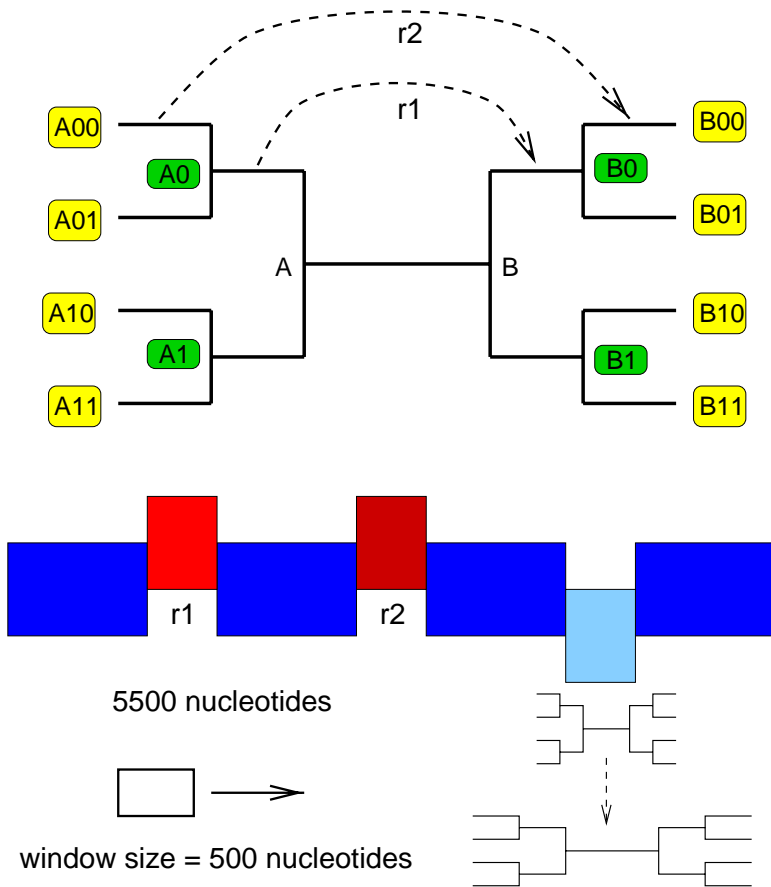
Bottom-up **average linkage clustering** better performance than top-down (K-means) clustering in terms of **complexity versus information content**.

Assessment of the method

Data

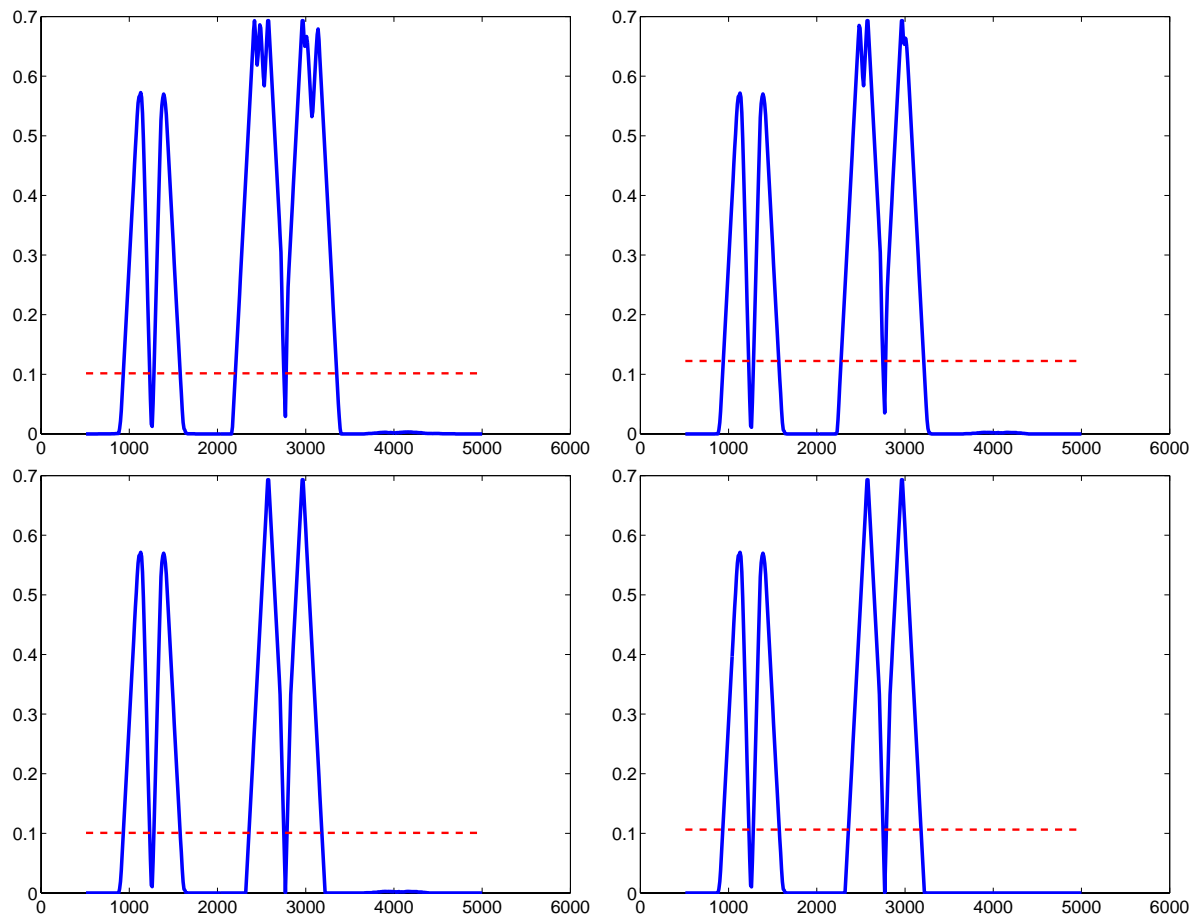
- Simulated recombination (8 sequences)
- Hepatitis B virus (10 sequences)
- Maize actin genes (8 sequences)

Simulated Data

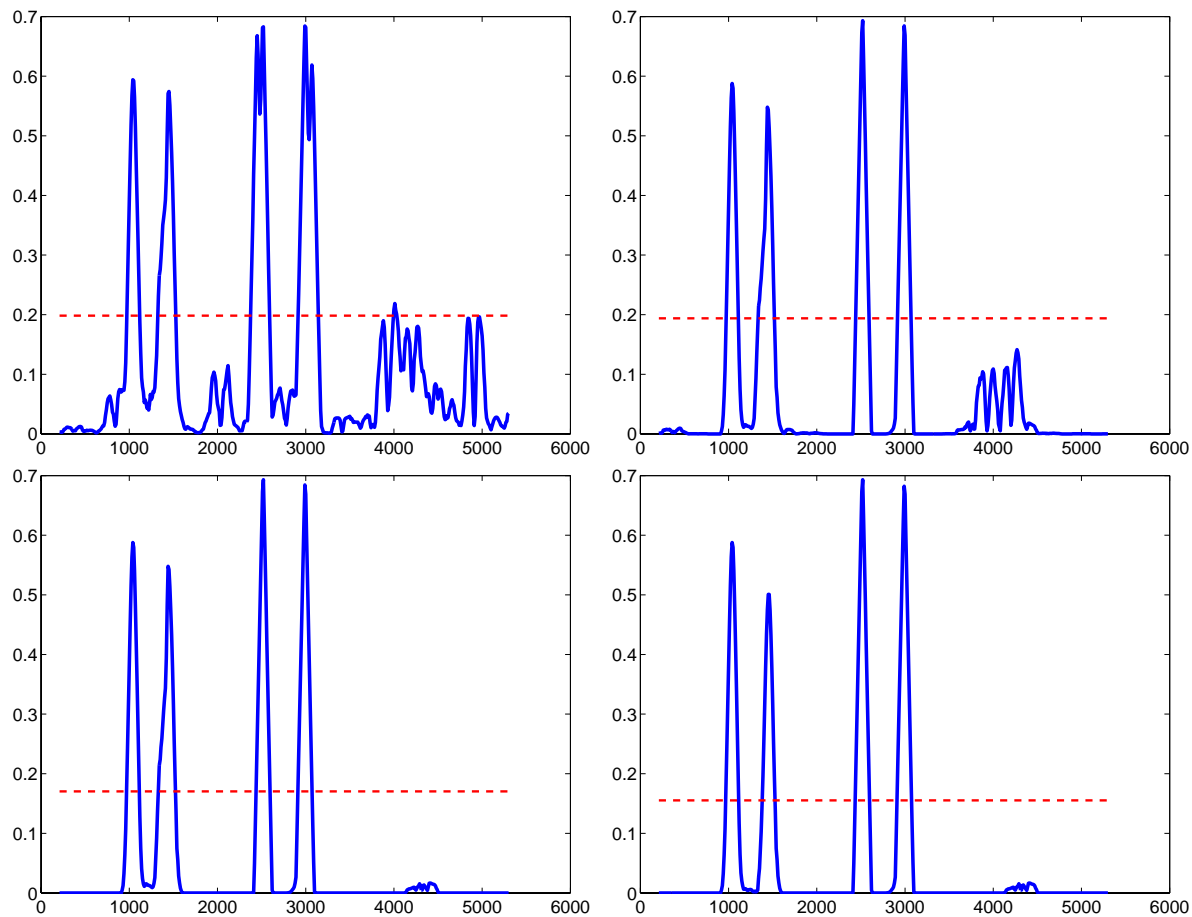


Sequence divergence: $w = 0.1$, $w = 0.01$. Window size: 500, 200

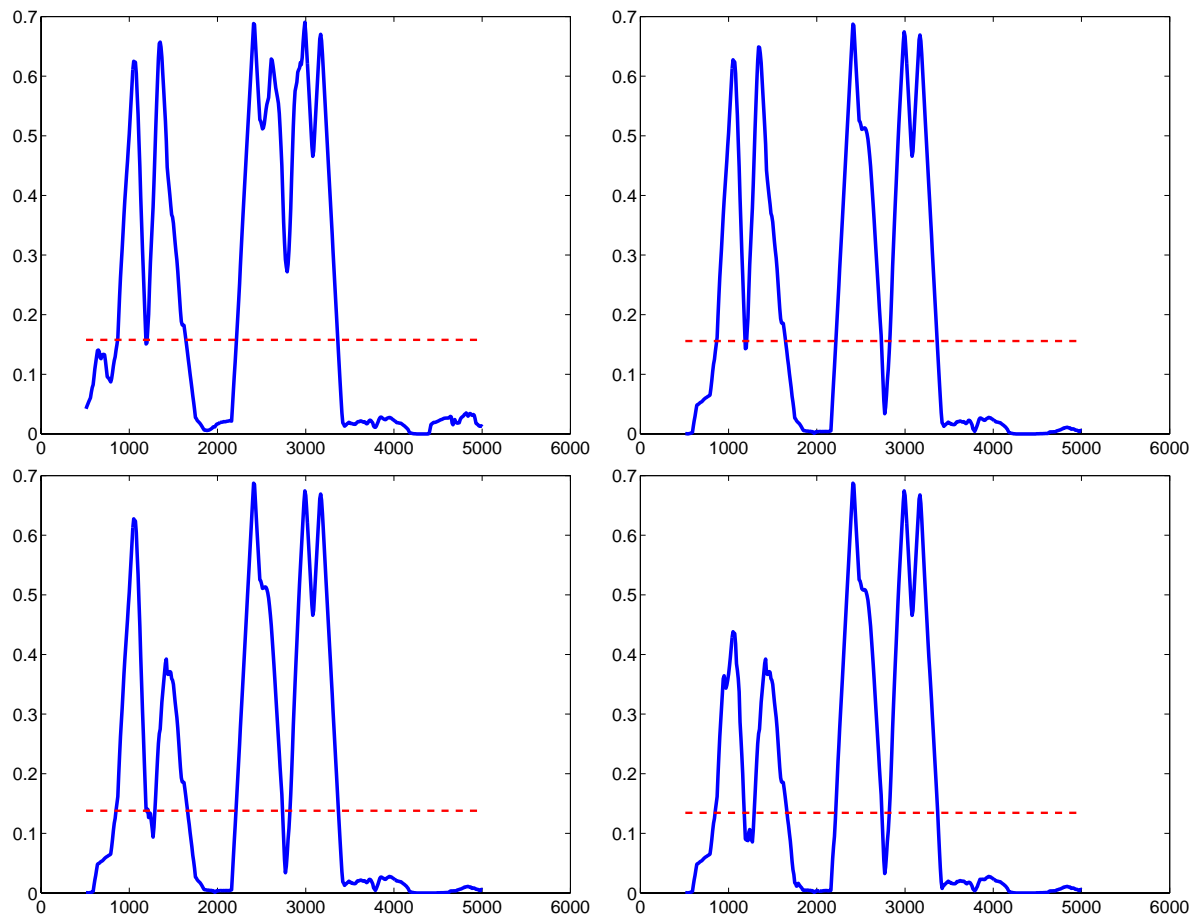
Sequence divergence: high ($w = 0.1$), window=500, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$



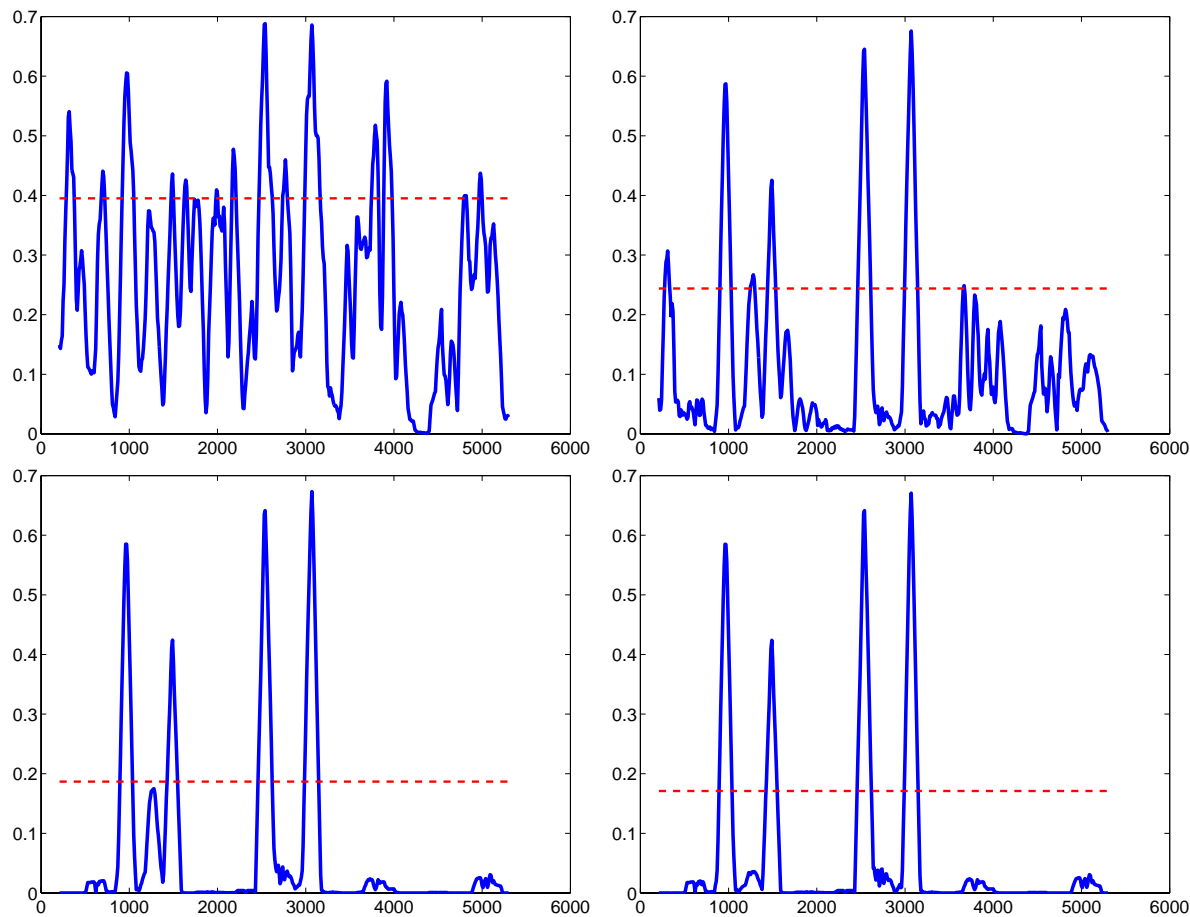
Sequence divergence: high ($w = 0.1$), window=200, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$



Sequence divergence: low ($w = 0.01$), window=500, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$



Sequence divergence: low ($w = 0.01$), window=200, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$

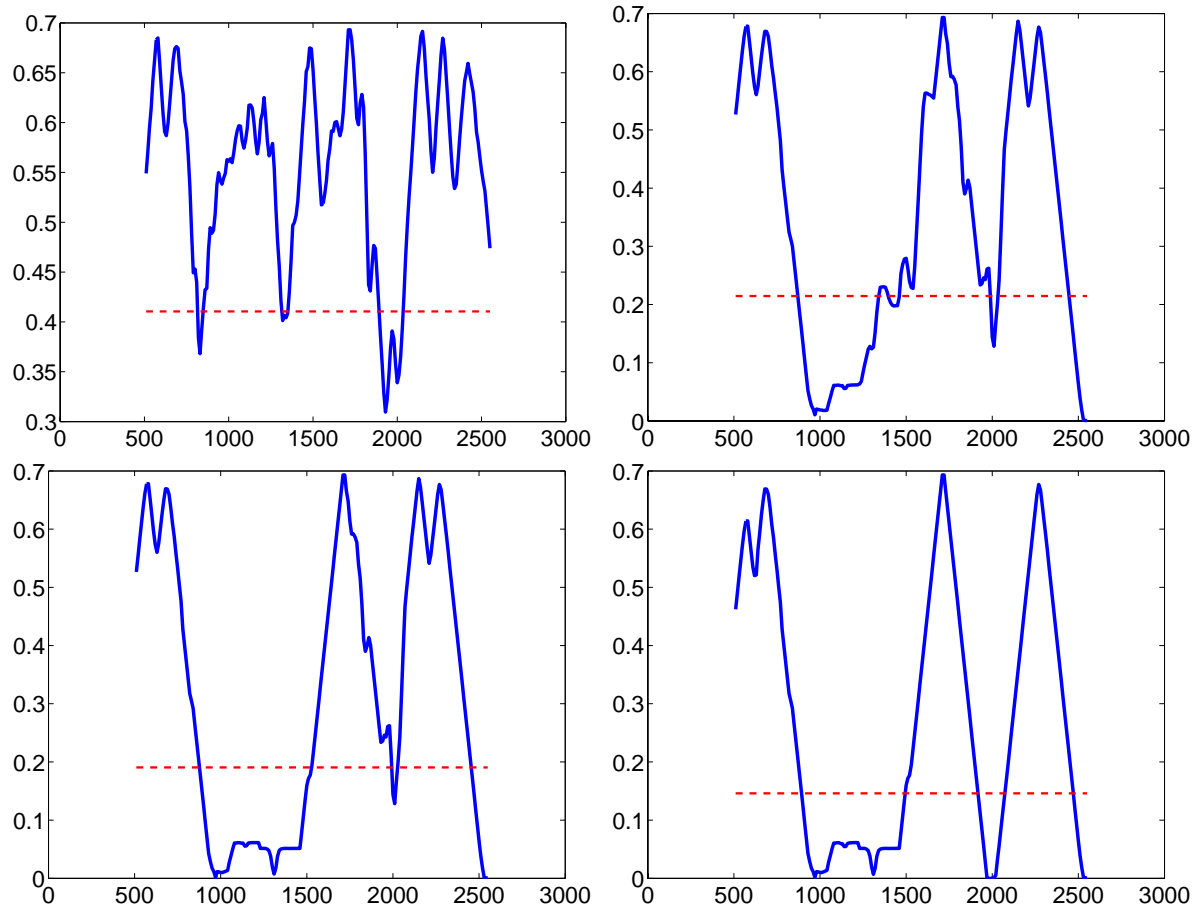


Hepatitis B virus

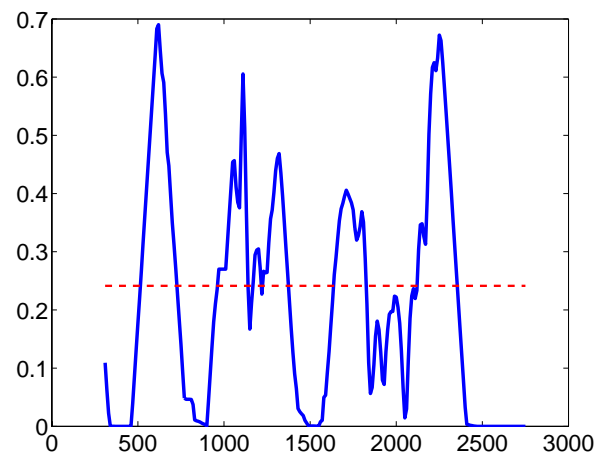
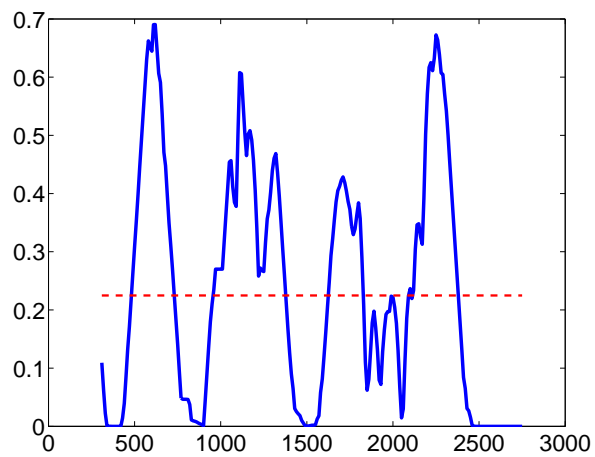
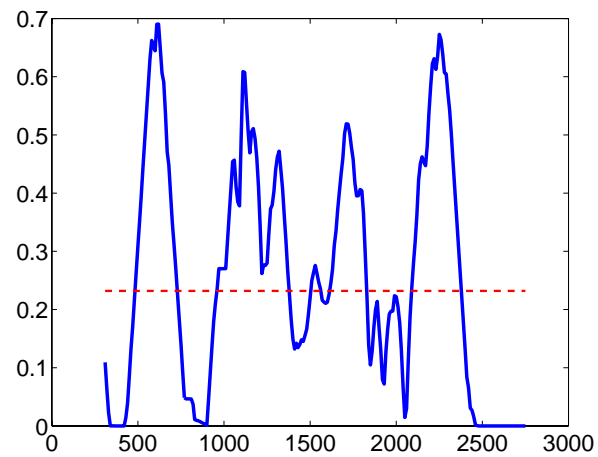
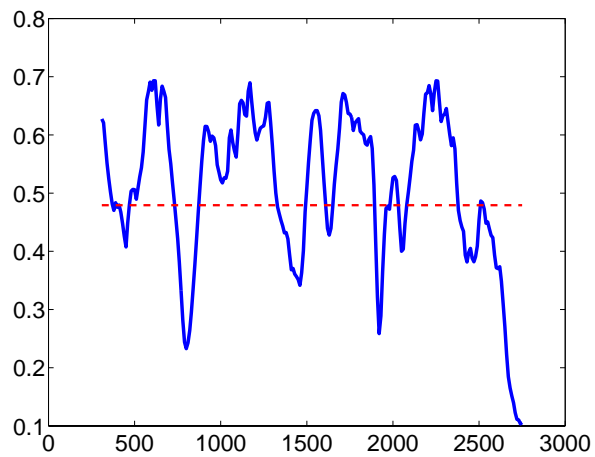
10 strains, 3049 bp

Bollyky, Rambaut, Harvey, Holmes (1996)
Journal of Molecular Evolution 42, 97-102

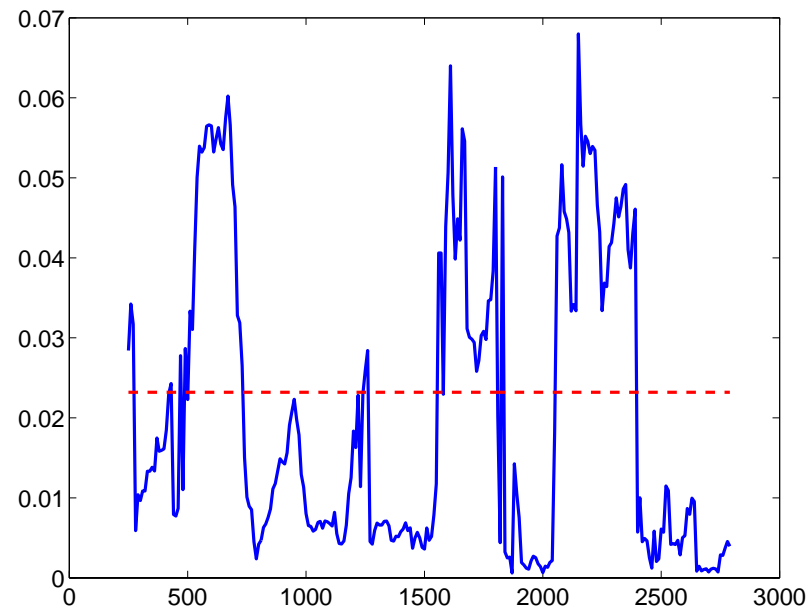
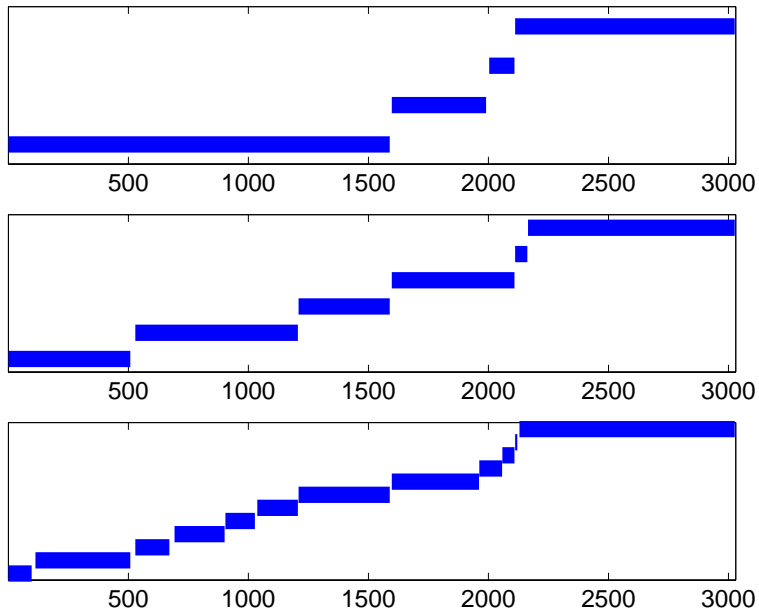
Data: Hepatitis-B, window=500, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$



Data: Hepatitis-B, window=300, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$



Left: Recpars (20, 10, 5). Right: DSS (win=300)



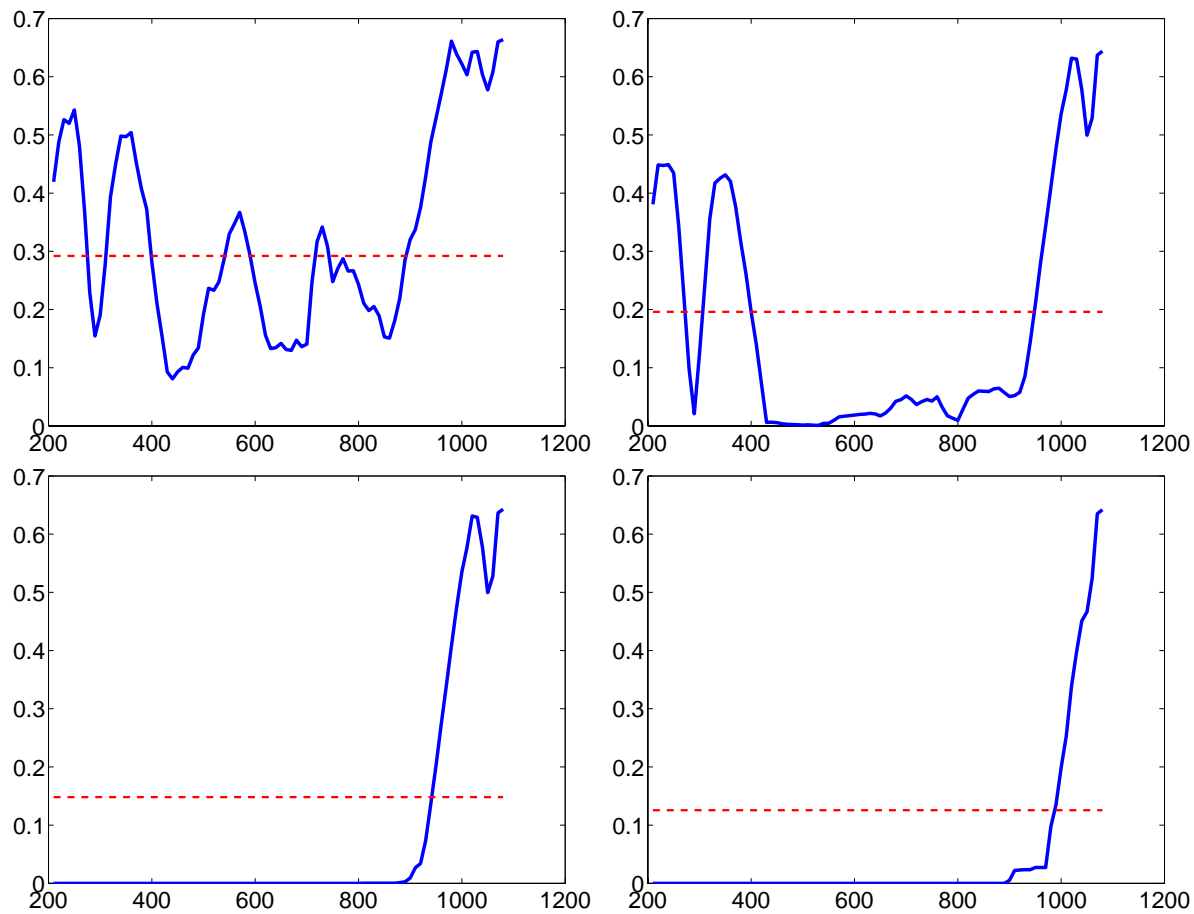
Maize actin genes

8 strains, 1281 bp

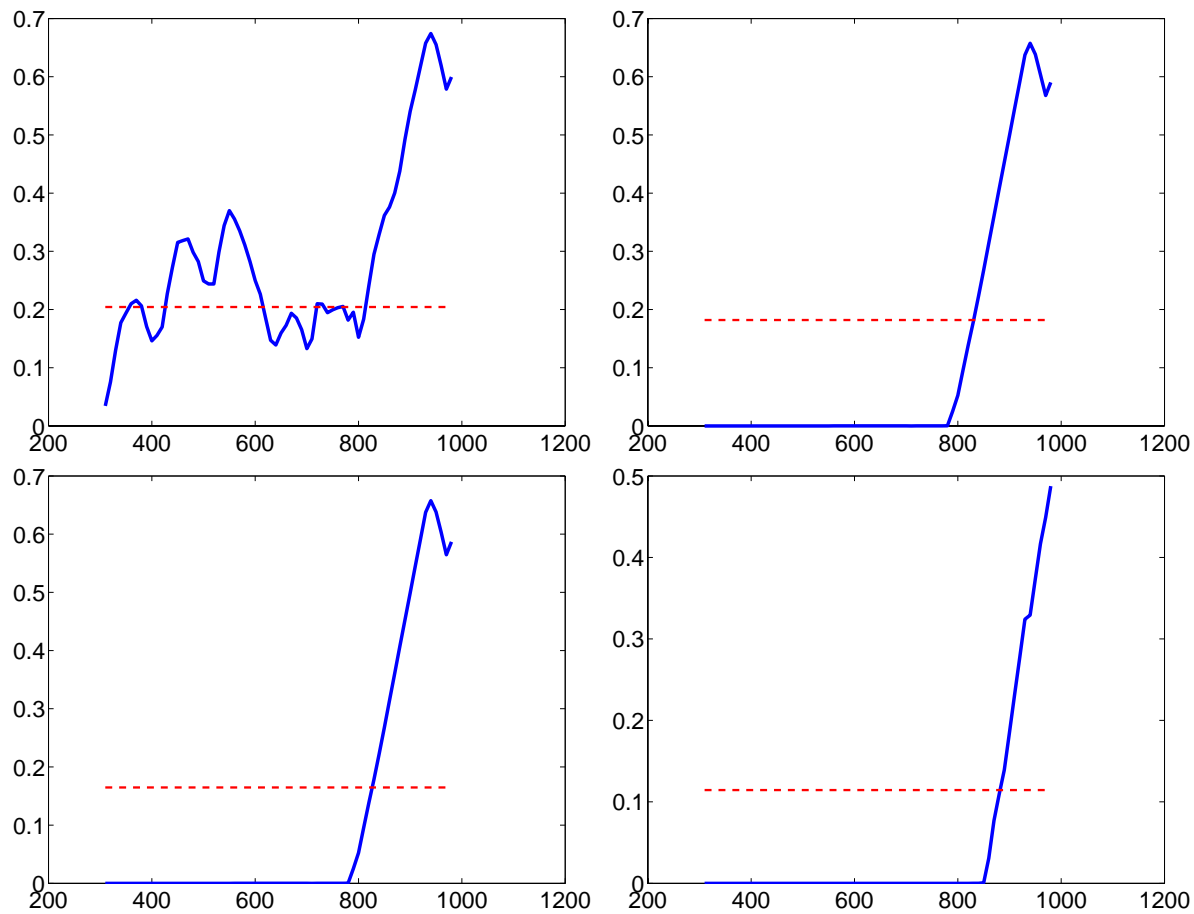
Moniz de Sa, Drouin (1996)

Molecular Biology and Evolution 13, 1198-1212

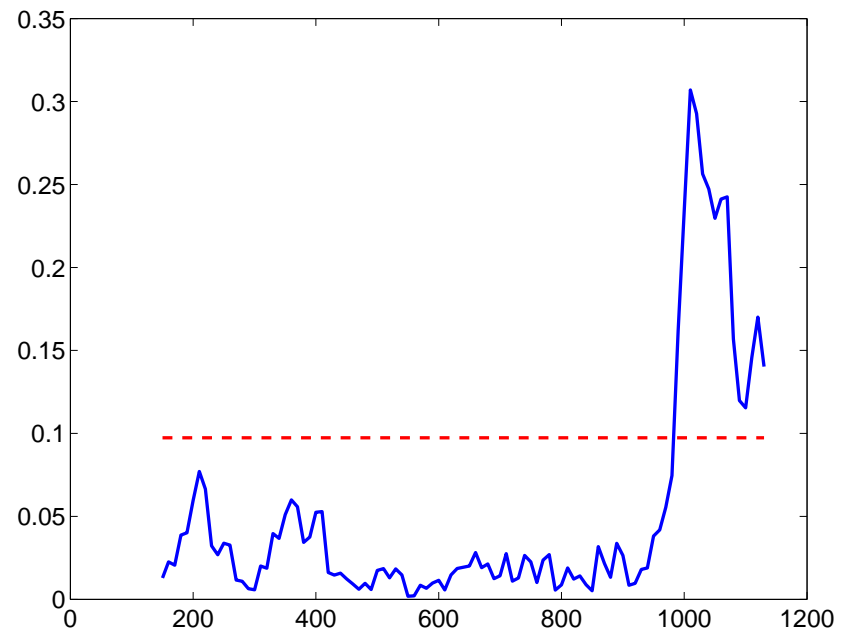
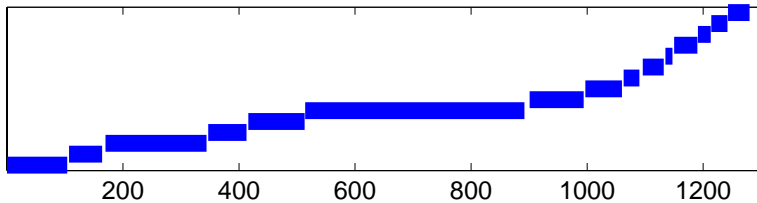
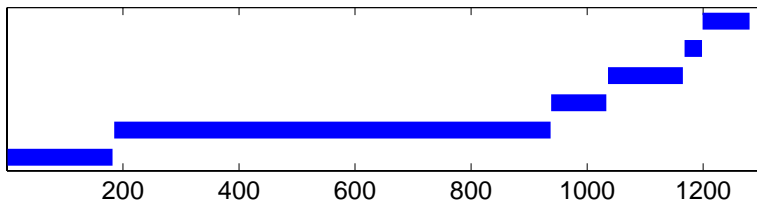
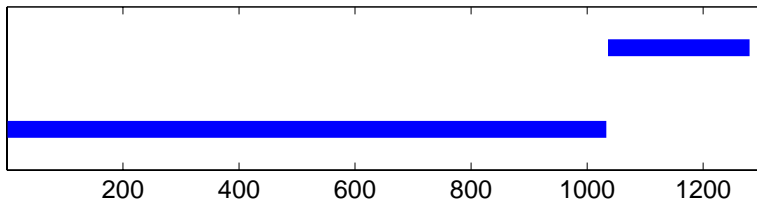
Data: Maize actin genes, window=200, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$



Data: Maize actin genes, window=300, N-topologies: $\begin{bmatrix} \infty & 7 \\ 5 & 3 \end{bmatrix}$



Left: Recpars (10, 5, 3). Right: DSS (win=300)



Conclusions

- PDM method = Window-based phylogenetic method
- Likelihood based → Less information loss than DSS (distance method) and RecPars (parsimony)
- Problem of diffuse posterior distribution → solved by pruning.
- Pruning → moderate reduction of window size possible → improved spatial resolution.
- Number of cluster K related to prior knowledge.
- Simulations: Similar results for $K = 3, 5, (7)$ → Robust with respect to moderate variation in K .