# Learning with Admixture: Modeling, Optimization, and Applications in Population Genetics


| | |
|---|---|
| PhD Student: | Jade Y. Cheng |
| Main supervisor: | Dr. Thomas Mailund |
| Co-supervisor: | Dr. Christian N. S. Pedersen |
| Main institution: | Bioinformatics Research Centre,<br>Department of Computer Science,<br>Aarhus University, Denmark |
| Visiting supervisor: | Dr. Rasmus Nielsen |
| Visiting institution: | Center for Theoretical Evolutionary Genomics,<br>Genetics and Statistics,<br>University of California, Berkeley, USA |

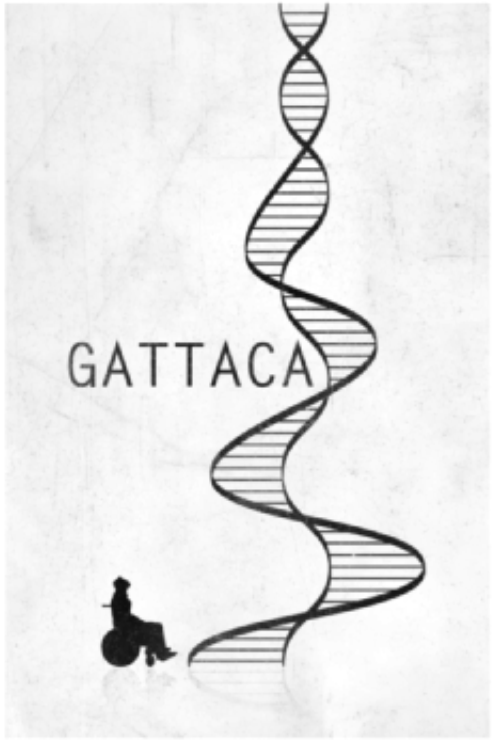
# Population Genetics

A branch of applied mathematics

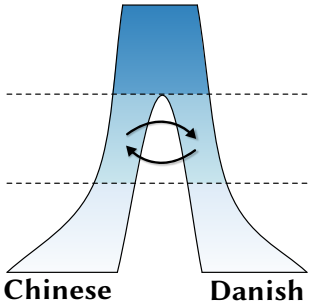
A translation of scientific **observations** into mathematical models

To produce quantitative **predictions** about evolution

**Observations**

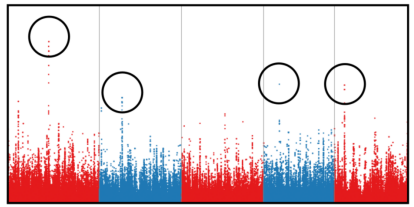


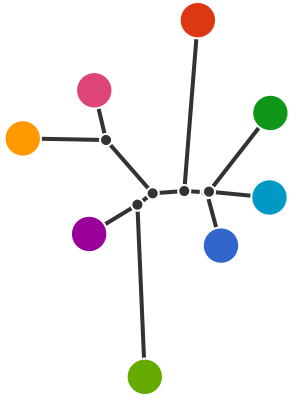

**Predictions**




Demographics

Selection



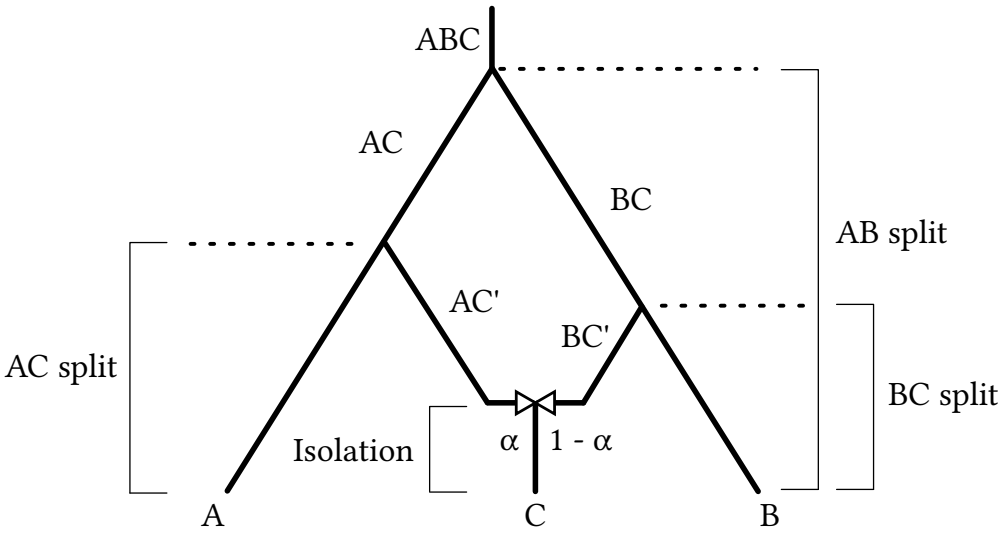


Population tree

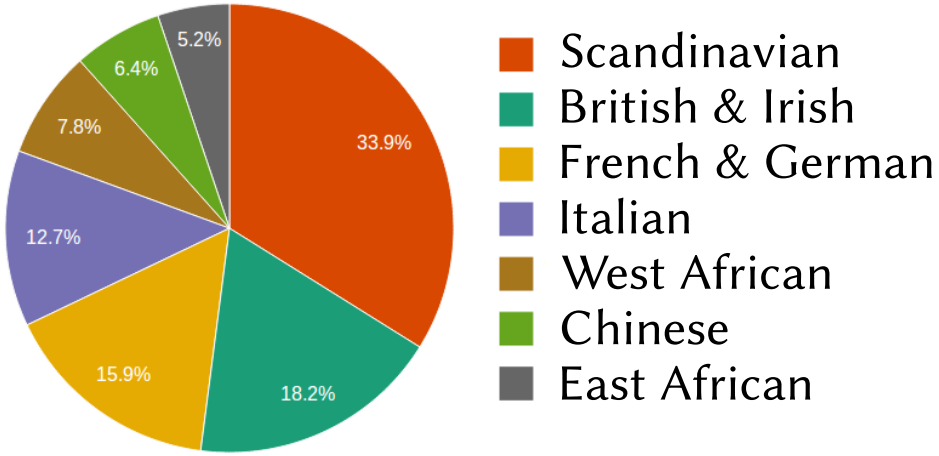
Population structure

# Admixture

Admixture, gene flow, and hybridization
Important forces in shaping evolutionary history



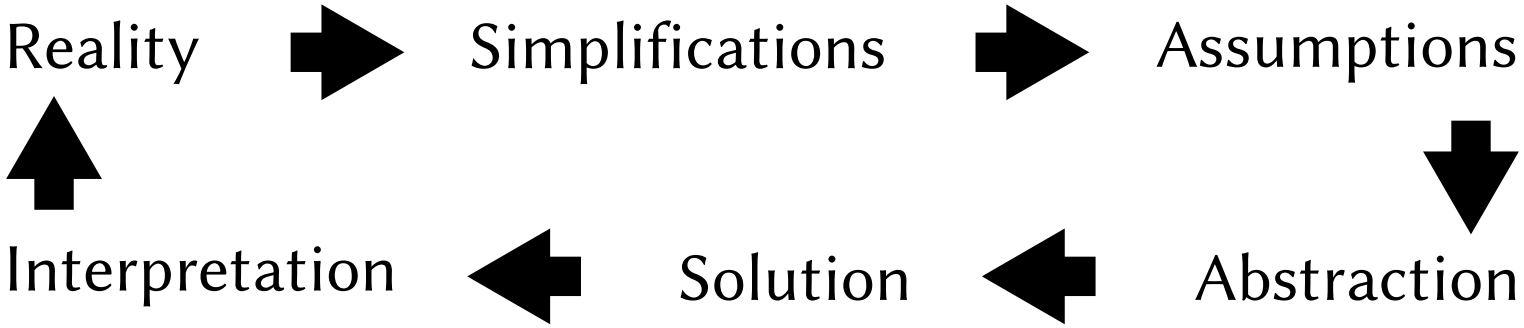

Admixture event




Admixture proportions

# Modeling and Optimization

A mathematical model is an abstraction that describes portions of reality
It is expressed in the language of mathematics




Numerical optimization finds values of variables that optimize an objective
No universal algorithm, each is designed for a particular type of problem

| | | Emphasis of this dissertation |
|---|---|---|
| Discrete | vs. | Continuous optimization |
| Stochastic | vs. | Deterministic optimization |
| Unconstrained | vs. | Constraint optimization |

# Project history and me

CoalHMM

The development of CoalHMM at BiRC dates back to 2007.

I joined Dr. Thomas Mailund's development team in early 2014.

I have mainly contributed in two directions:

1. Optimization through heuristic-based evolutionary algorithms
2. Modeling and inferring historical admixture events

Ohana

Ohana started at UC Berkeley with Dr. Rasmus Nielsen, early 2015.

I named the project Ohana, meaning “family” in Hawaiian.

I am the main contributor for all methods and software modules:

1. Admixture inference
2. Evolutionary tree estimation
3. Selection identification



# Publications

## CoalHMM

**Jade Yu Cheng** and Thomas Mailund. 2015 "Ancestral population genomics using coalescence hidden Markov models and heuristic optimisation algorithms" *Computational Biology and Chemistry*, 57, pp.80-92

**Jade Yu Cheng** and Thomas Mailund. 2016 "A coalescent hidden Markov model for inferring admixture relationships" *in preparation*

Tianying Lan, **Jade Yu Cheng**, Aakrosh Ratan, Webb Miller, Stephan C. Schuster, Karyn Rode, Todd Atwood, Sean Farley, Dick Richard T. Shideler, Sandra L. Talbot, Thomas Mailund, Charlotte Lindqvist. 2016 "Genome-wide evidence for a hybrid origin of modern polar bears" *bioRxiv, p.047498*

## Ohana

**Jade Yu Cheng**, Thomas Mailund, and Rasmus Nielsen. 2016 "Ohana, a tool set for population genetic analyses of admixture components" *bioRxiv, p.071233, submitted to Bioinformatics*

Anna-Sapfo Malaspinas, Michael C Westaway, Craig Muller, Vitor C Sousa, Oscar Lao, Isabel Alves, Anders Bergström, Georgios Athanasiadis, **Jade Y Cheng**, ..., Eske Willerslev. 2016 "The genomic history of Australia" *Nature*, doi:10.1038/nature18299

Georgios Athanasiadis, **Jade Yu Cheng**, Bjarni J. Vilhjálmsson, Frank Grønlund Jørgensen, Thomas D. Als, Stephanie Le Hellard, Thomas Espeseth, Patrick F. Sullivan, Christina M. Hultman, Peter Kjærgaard, Mikkel Heide Schierup, Thomas Mailund. 2016 "Nationwide genomic study in Denmark reveals remarkable population homogeneity" *to appear in Genetics*

# Admixture CoalHMM



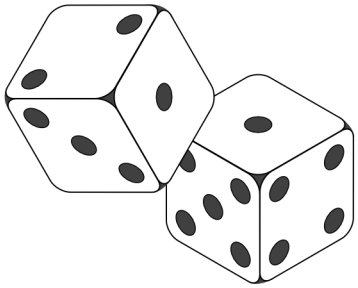
# Introduction to CoalHMM

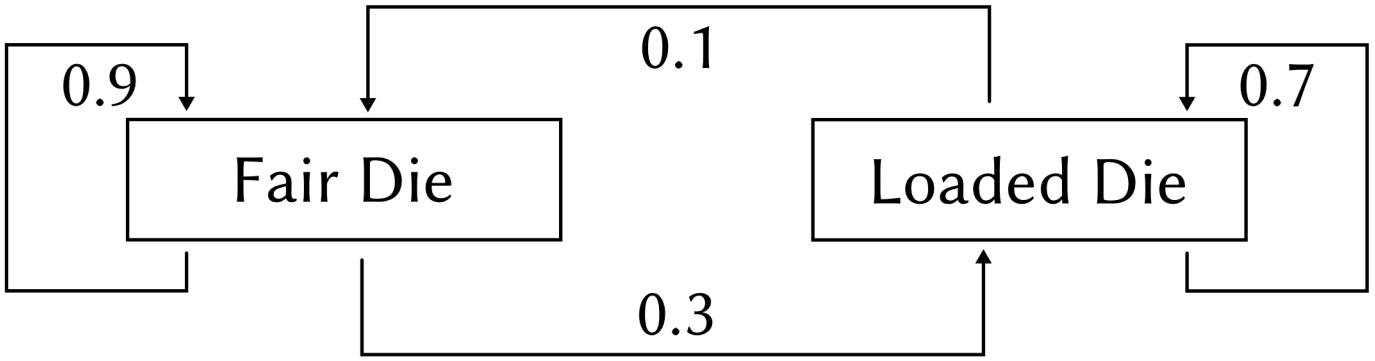
| | |
|---|---|
| Model: | Hidden Markov model |
| Foundation: | Coalescence theory |
| Key assumption: | Local genealogy is Markovian along the alignment |
| Parametrized: | All forms of gene flow<br>Population splits<br>Coalescent rate<br>Recombination rate |
| Similar methods: | PSMC leads the way in popularity |
| Complexity: | Exponential as samples increase |



# HMM Example

The Occasionally Dishonest Casino:






Observations:

12455264621461461361366616646616366163661511514612356234 4

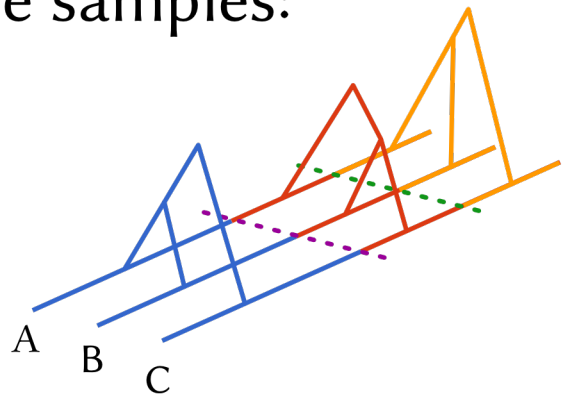
Questions:

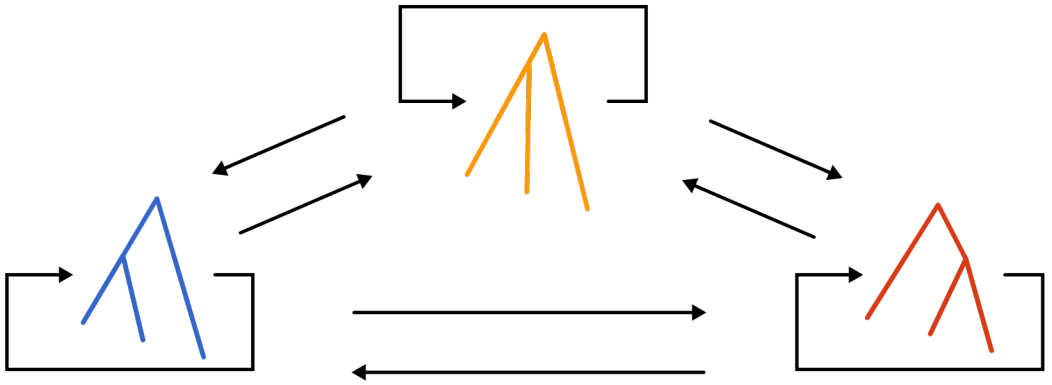
How likely is this sequence, given the model, i.e. how they cheat?
How "loaded" is the loaded die? How often does the die change?
What portion of the sequence was generated with each die?

Reference: Machine Learning 10-701/15-781, Computer Science, Carnegie Mellon University

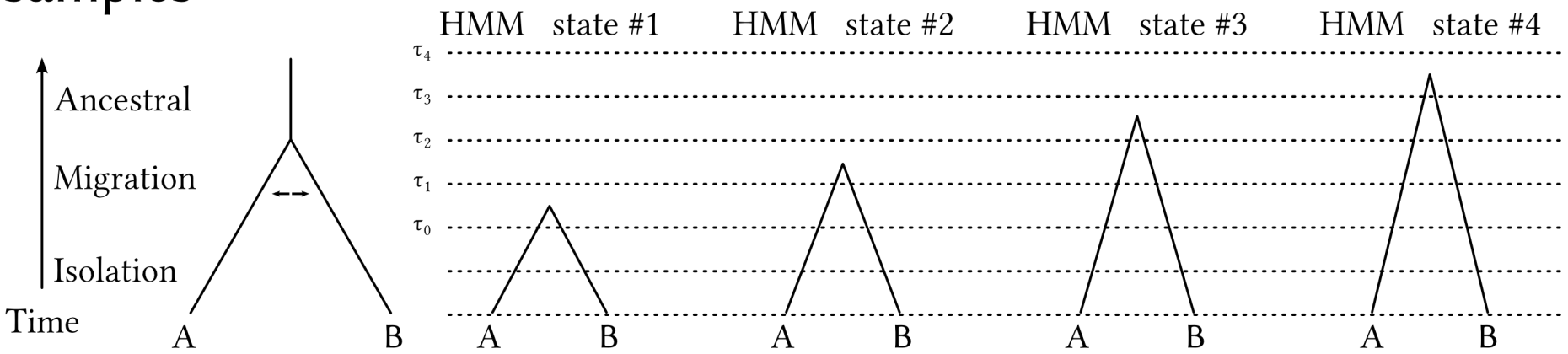
# CoalHMM–HMM

Three samples:



Two samples:



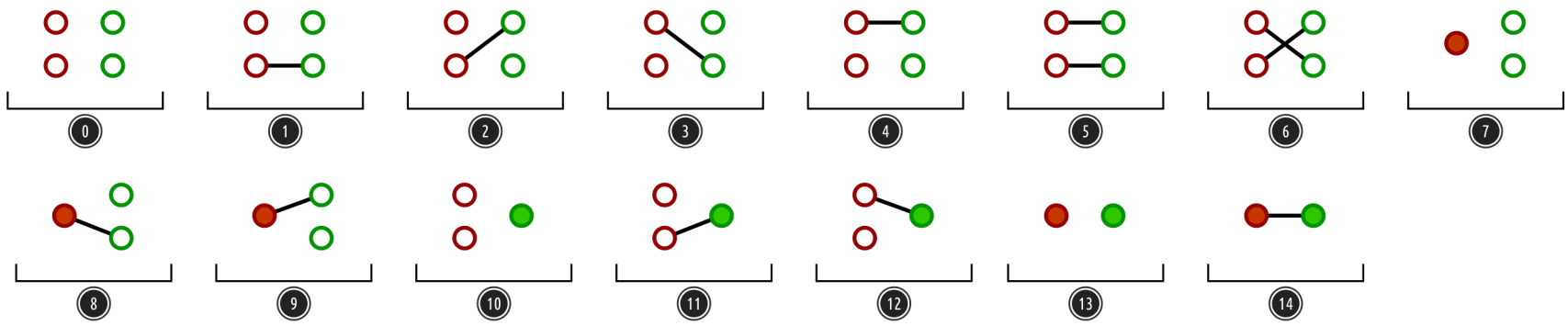

Observations:

```
ACT-ACTGACTTGACTGACTTGACTCACTTGACTGACG--CTGG...
ACTGACTGACTTGACTG-CTTGTCTGACTAGACTGACGAGGAGG...
→ 000200000000000020000100100000000000221100...
```

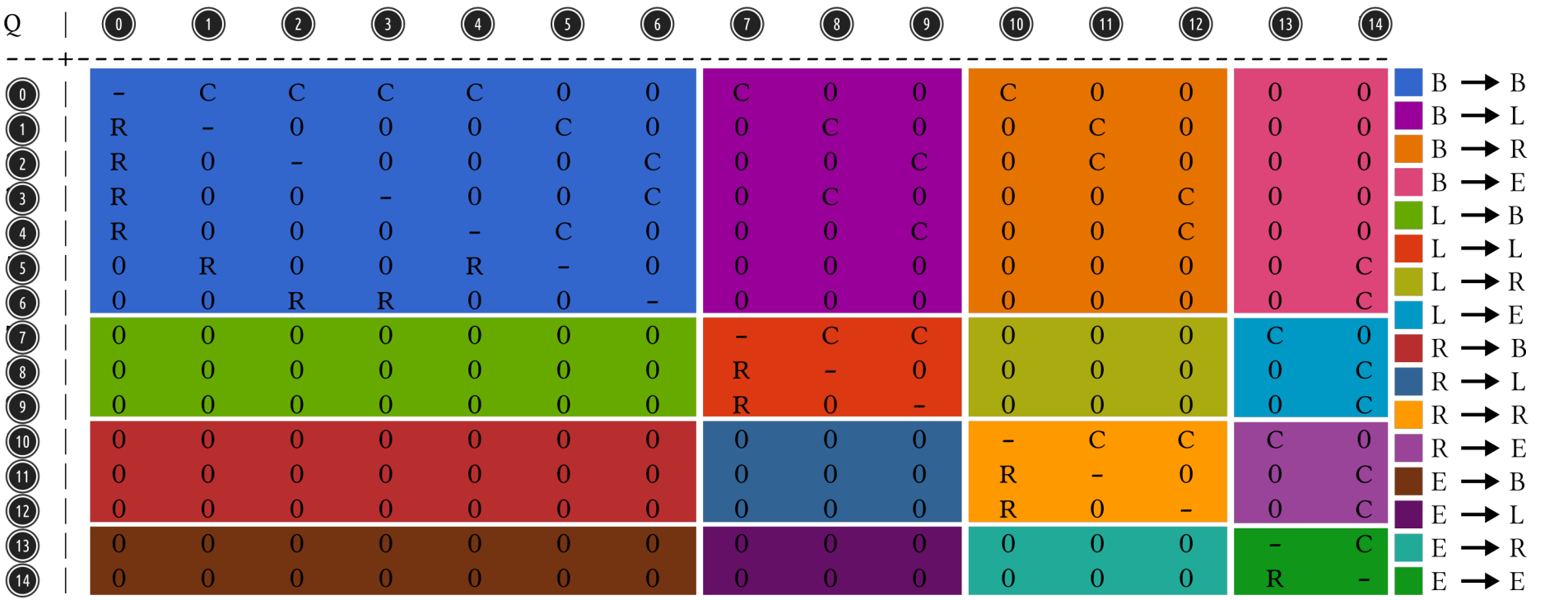
# CoalHMM–CTMC

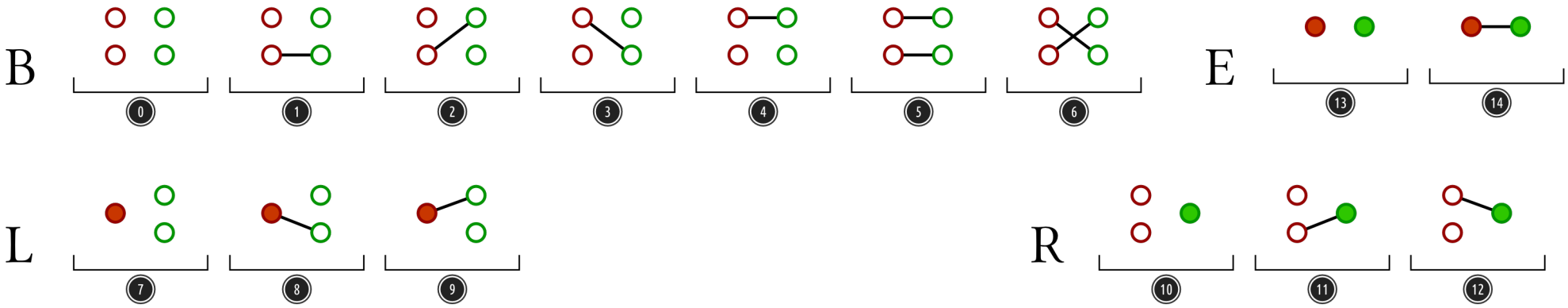
Markovian along the alignment, so only consider two adjacent nucleotides

| Q | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | – | C | C | C | C | 0 | 0 | C | 0 | 0 | C | 0 | 0 | 0 | 0 |
| 1 | R | – | 0 | 0 | 0 | C | 0 | 0 | C | 0 | 0 | C | 0 | 0 | 0 |
| 2 | R | 0 | – | 0 | 0 | 0 | C | 0 | 0 | C | 0 | C | 0 | 0 | 0 |
| 3 | R | 0 | 0 | – | 0 | 0 | C | 0 | C | 0 | 0 | 0 | C | 0 | 0 |
| 4 | R | 0 | 0 | 0 | – | C | 0 | 0 | 0 | C | 0 | 0 | C | 0 | 0 |
| 5 | 0 | R | 0 | 0 | R | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | C |
| 6 | 0 | 0 | R | R | 0 | 0 | – | 0 | 0 | 0 | 0 | 0 | 0 | 0 | C |
| 7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | C | C | 0 | 0 | 0 | C | 0 |
| 8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R | – | 0 | 0 | 0 | 0 | 0 | C |
| 9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R | 0 | – | 0 | 0 | 0 | 0 | C |
| 10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | C | C | C | 0 |
| 11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R | – | 0 | 0 | C |
| 12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R | 0 | – | 0 | C |
| 13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | – | C |
| 14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | R | – |



# CoalHMM—Joint Probability



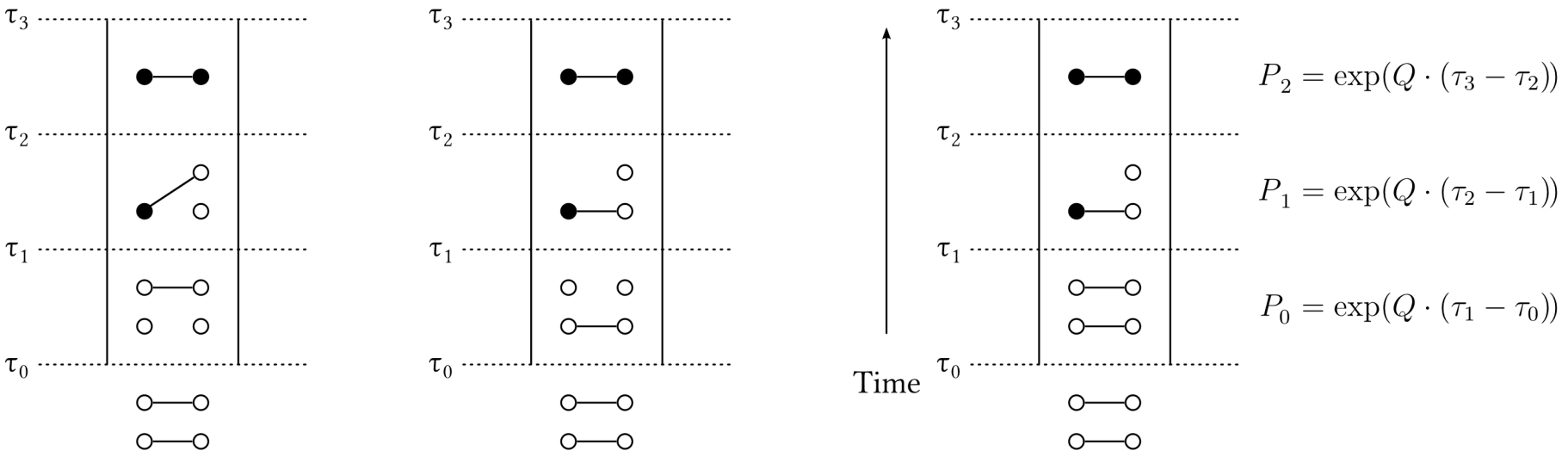



# CoalHMM—Joint Probability

$$J_{ij} = \begin{cases} \sum_\alpha \sum_\beta M_{\alpha\beta} & \text{when } i \leq j \\ J_{ji} & \text{when } i > j \end{cases}$$

$$M_{ij} = \begin{cases} (P_0)_{0B} \times \cdots \times (P_{i-1})_{BB} \times (P_i)_{BL} \times (P_{i+1})_{LL} \times \cdots \times (P_{j-1})_{LL} \times (P_j)_{LE} & \text{when } i < j \\ (P_0)_{0B} \times (P_1)_{BB} \times \cdots \times (P_{i-1})_{BB} \times (P_i)_{BE} & \text{when } i = j \end{cases}$$

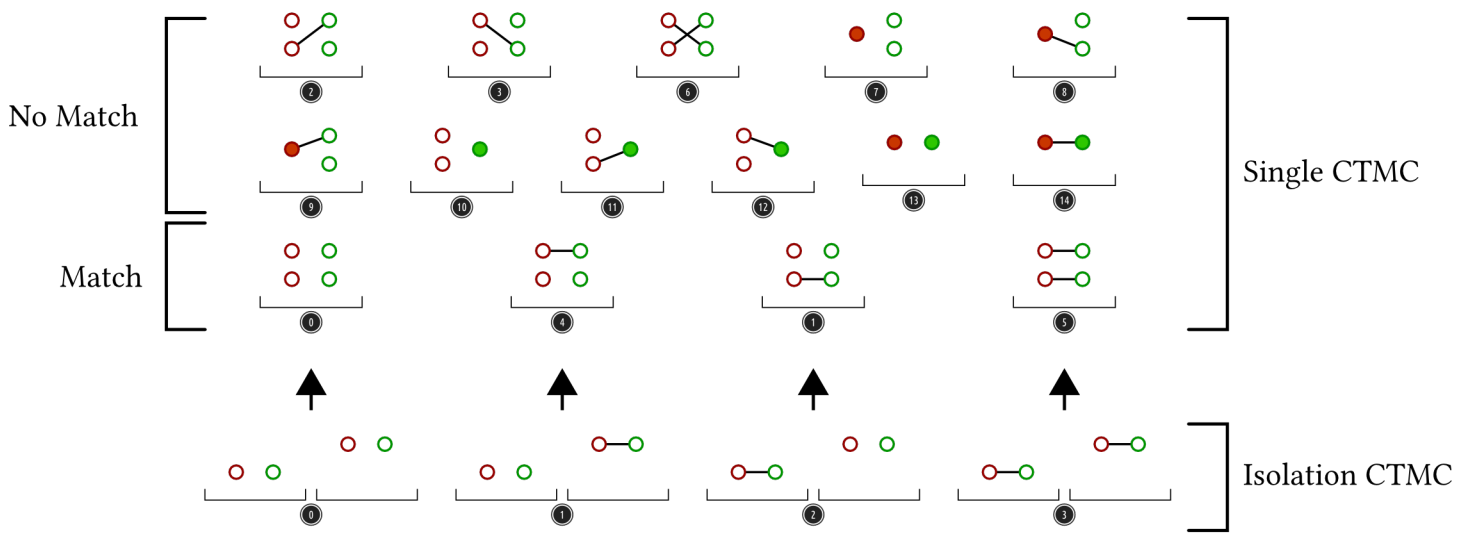
## Example:

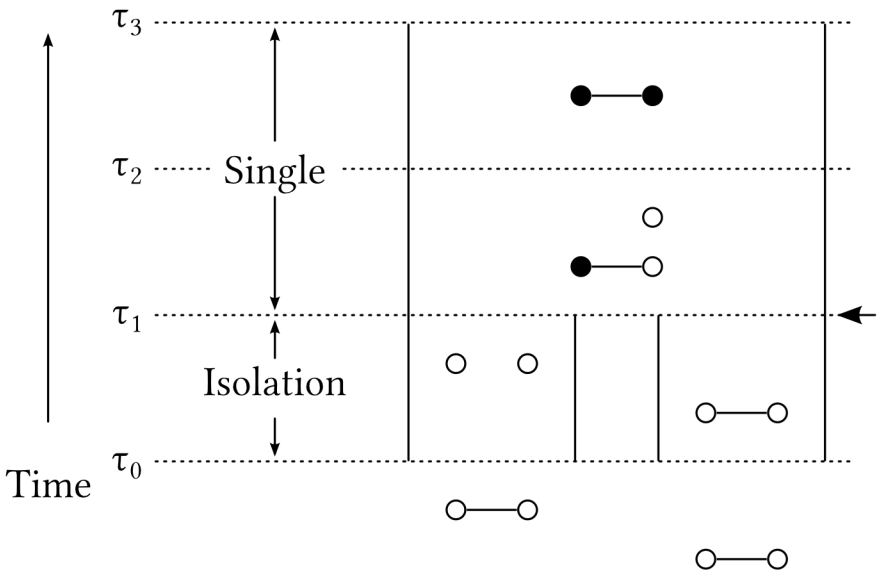



$$J_{23} = \sum_\alpha \sum_\beta ((P_0)_{0B} \times (P_1)_{BL} \times (P_2)_{LE})_{\alpha\beta}$$

# CoalHMM–CTMC Projection

Example:



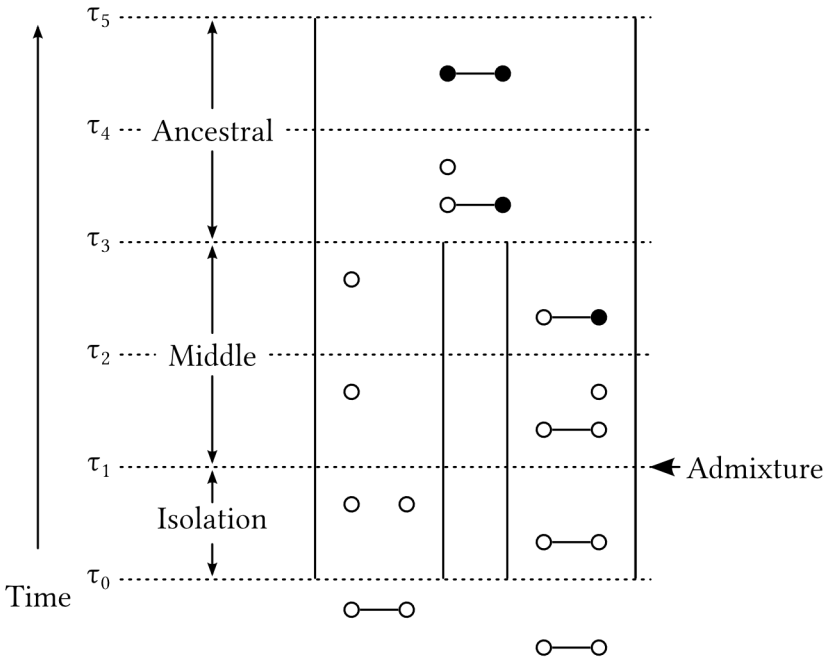




$P_{\text{isolation}\to\text{single}}$

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

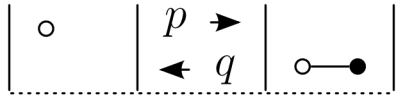
# CoalHMM—Admixture

## Example:




Source state:

```
a. {1, ({1},{})} {2, ({2},{1,2})}
```




Destination states:

```
a. {1, ({1},{})} {2, ({2},{1,2})}
```

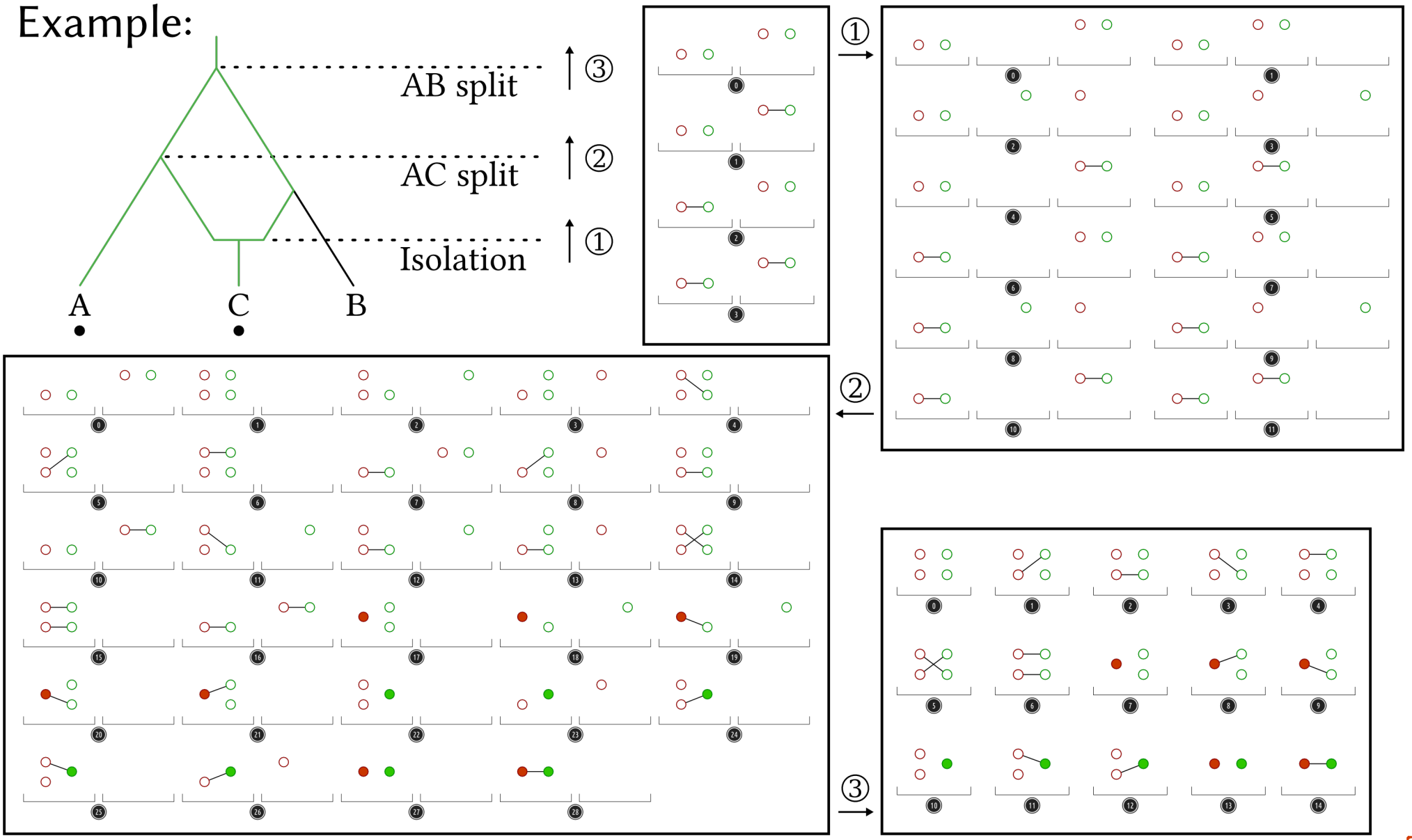
```
b. {1, ({1},{})} {1, ({2},{1,2})}
```

```
c. {2, ({1},{})} {2, ({2},{1,2})}
```

```
d. {2, ({1},{})} {1, ({2},{1,2})}
```

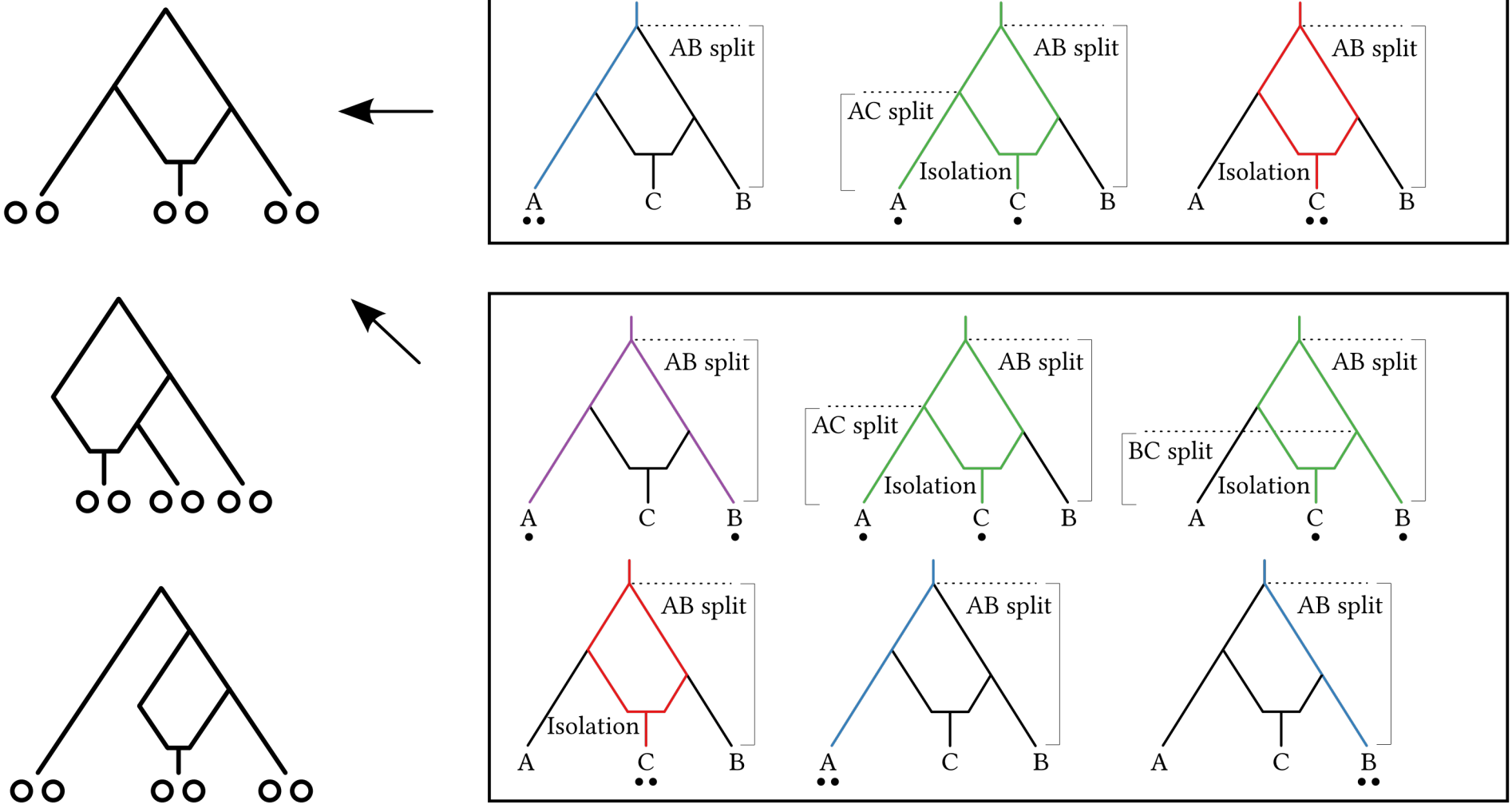
| binary index | pieces | destination | probability |
|---|---|---|---|
| 00 | nobody moves | a | $(1-p)\cdot(1-q)$ |
| 01 | left piece stays; right piece moves | b | $(1-p)\cdot q$ |
| 10 | left piece moves; right piece stays | c | $p\cdot(1-q)$ |
| 11 | both pieces move | d | $p\cdot q$ |

# CoalHMM–HMM



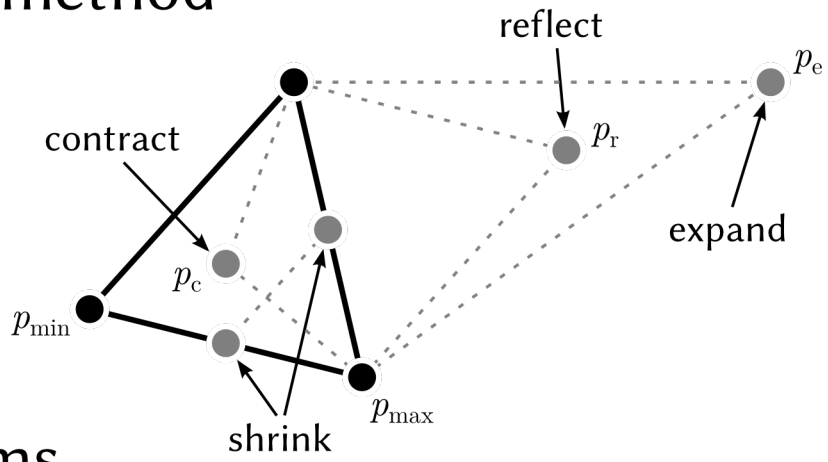
# CoalHMM—Composite Likelihood

Example:



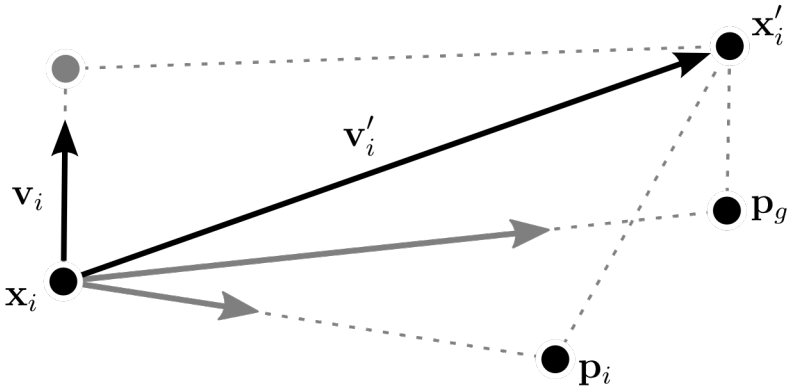
# CoalHMM—Optimization

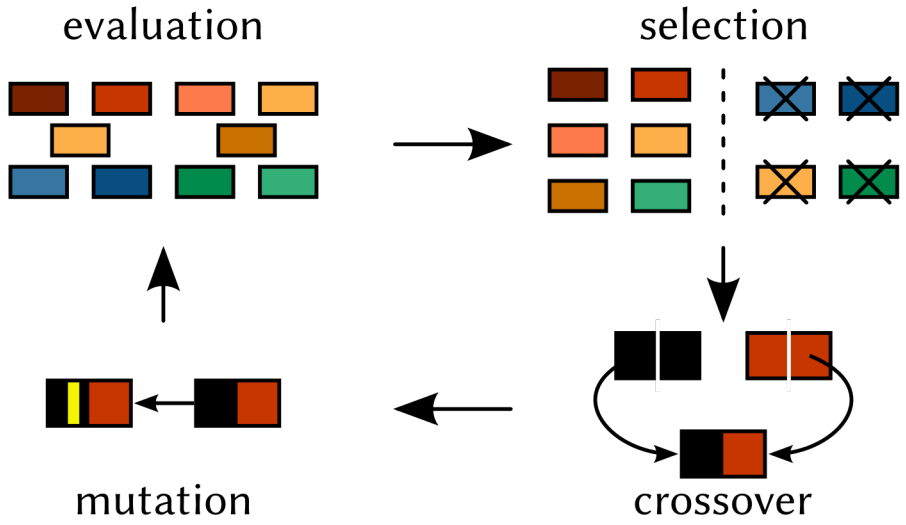
## Nelder-Mead simplex method



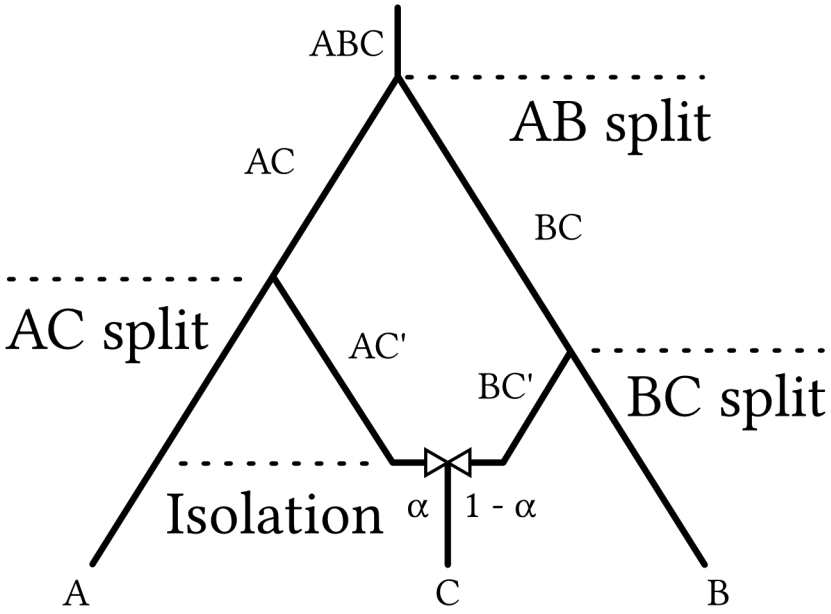

## Evolutionary algorithms

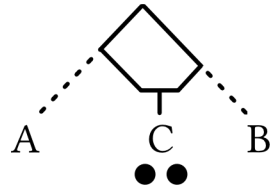
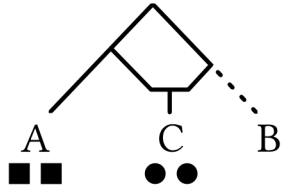
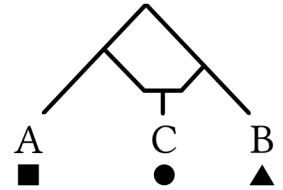
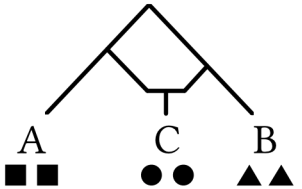
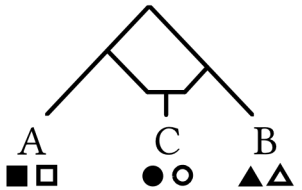
## Particle swarm optimization (left) and Genetic algorithm (right)



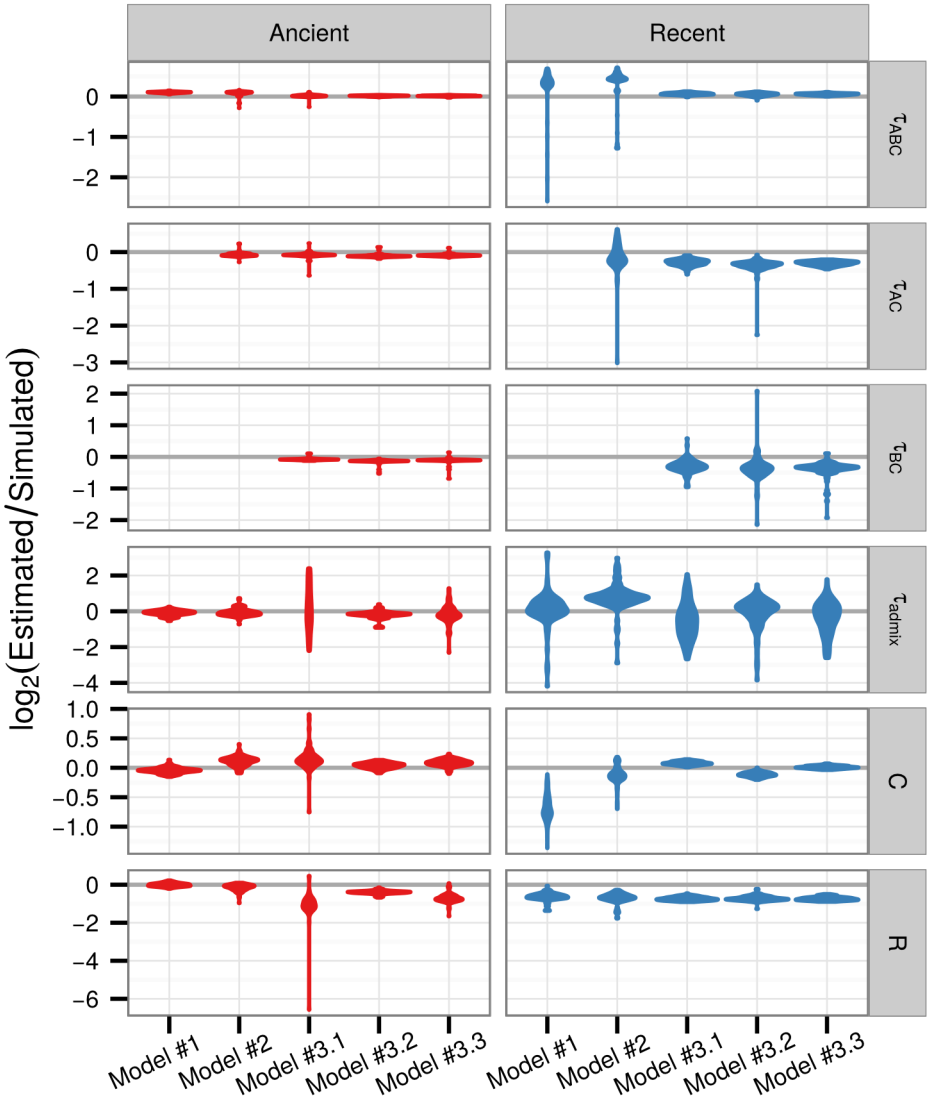



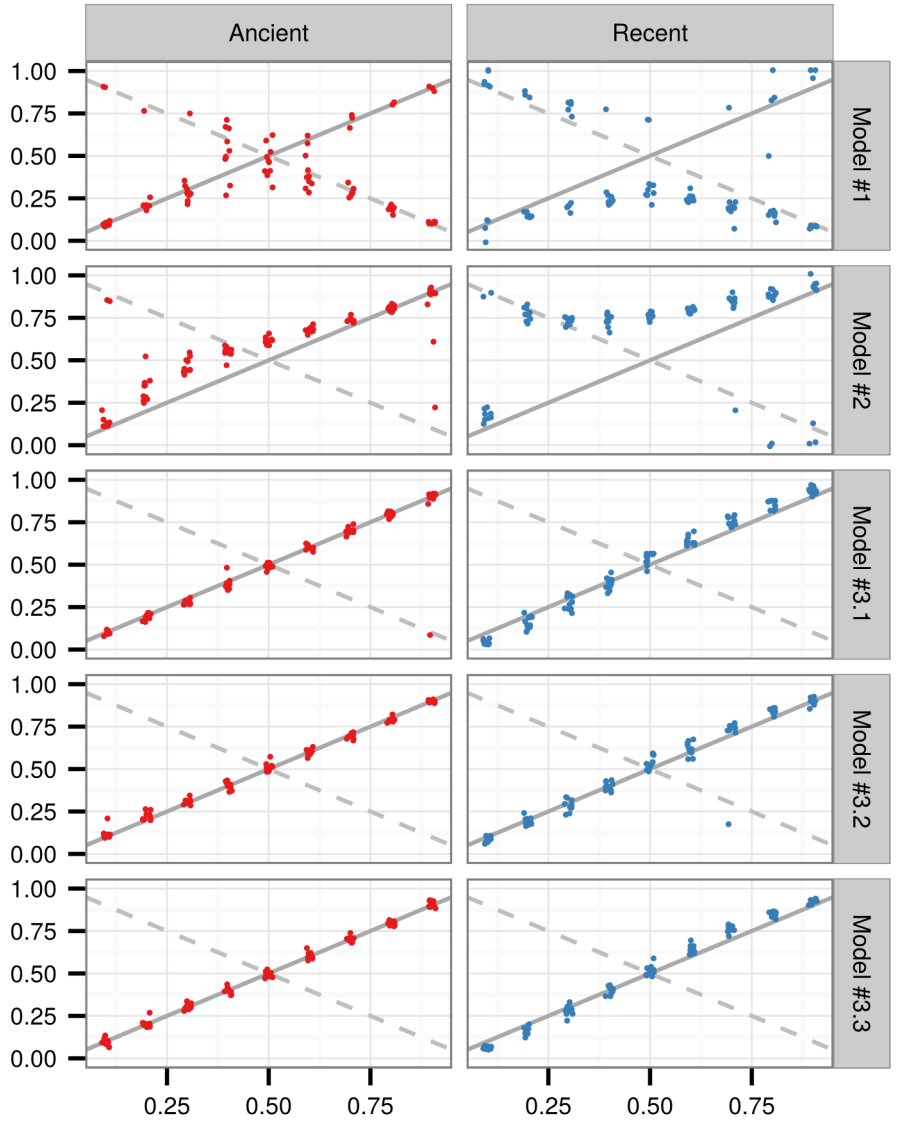
# CoalHMM—Simulation




| | A | B | C | Samples |
|---|---|---|---|---|
| #1 | | | ✓ |  |
| #2 | ✓ | | ✓ |  |
| #3-1 | ✓ | ✓ | ✓ |  |
| #3-2 | ✓ | ✓ | ✓ |  |
| #3-3 | ✓ | ✓ | ✓ |  |

# CoalHMM—Simulation



# Ohana



# Introduction to Ohana

Admixture module:

| | |
|---|---|
| Model: | Classical Structure model, Unsupervised learning |
| Optimization: | Sequential quadratic programming<br>Active Set algorithm<br>Complementarity Pivoting algorithm |

Evolutionary tree module:

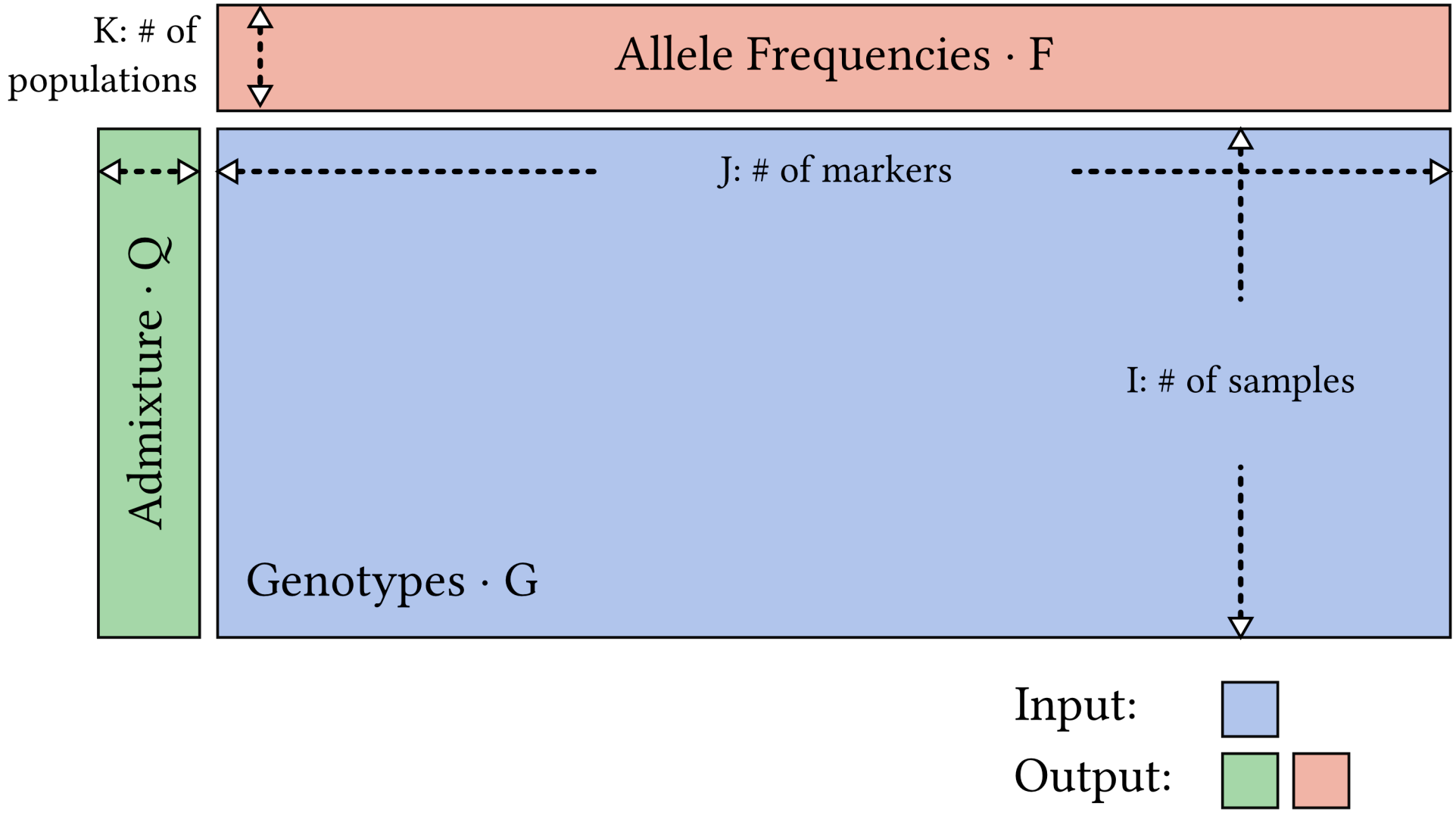
| | |
|---|---|
| Model: | Gaussian approximation |
| Optimization: | Nelder-Mead simplex optimization |

Selection module:

| | |
|---|---|
| Model: | Nested likelihood models |
| Optimization: | Nelder-Mead simplex optimization |



# Introduction to Ohana—Data



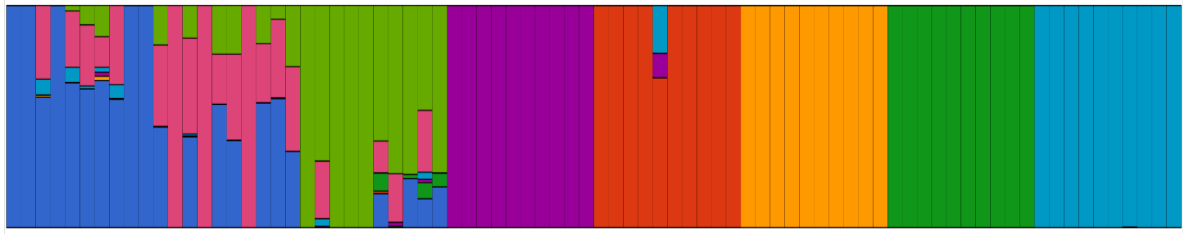
# Introduction to Ohana—Admixture

K: # of populations

Admixture · Q

I: # of samples

K: # of populations = # of colors



I: # of samples

# Ohana—Structure Model

## Genotype observations

$$\ln\left[P_1^O\left(Q,\ F\right)\right] = \sum_i^I \sum_j^J \left[g_{ij} \cdot \ln\left(A_{ij}\right) + \left(2 - g_{ij}\right) \cdot \ln\left(B_{ij}\right)\right]$$

## Genotype likelihoods

$$P_1^L\left(Q,\ F\right) = \sum_g \left[Pr\left(X \mid g\right) \cdot Pr\left(g \mid Q,\ F\right)\right]$$

$$Pr\left(X_{ij} \mid g_{ij}\right) = \begin{cases} g_{ij}^{AA} & \text{for } AA \\ g_{ij}^{Aa} & \text{for } Aa \text{ or } aA \\ g_{ij}^{aa} & \text{for } aa \end{cases}$$

$$A_{ij} = \sum_k^K q_{ik} \cdot f_{kj}$$

$$B_{ij} = \sum_k^K q_{ik} \cdot \left(1 - f_{kj}\right)$$

$$\ln\left[P_1^L\left(Q,\ F\right)\right] = \sum_i^I \sum_j^J \ln\left(g_{ij}^{AA} \cdot A_{ij}^2 + g_{ij}^{aa} B_{ij}^2 + g_{ij}^{Aa} \cdot 2A_{ij}B_{ij}\right)$$

# Ohana—Optimization

## Quadratic Programming and Newton's update

$$\frac{1}{2}\bar{x}^T Q\bar{x} + c^T\bar{x}$$

$$F_T\left(x_n + \Delta x\right) = F\left(x_n\right) + F'\left(x_n\right)\Delta x + \frac{1}{2}F''\left(x_n\right)\Delta x^2$$

## Sequential Quadratic Programming

$$\ln\left[P_1^O\left(Q,\ F\right)\right] = \sum_i^I \sum_j^J \left[g_{ij}\cdot\ln\left(A_{ij}\right) + \left(2 - g_{ij}\right)\cdot\ln\left(B_{ij}\right)\right]$$

$$\ln\left[P_1^L\left(Q,\ F\right)\right] = \sum_i^I \sum_j^J \ln\left(g_{ij}^{AA}\cdot A_{ij}^2 + g_{ij}^{aa}B_{ij}^2 + g_{ij}^{Aa}\cdot 2A_{ij}B_{ij}\right)$$

## Equality and inequality constraints

$$F \rightarrow \forall\Delta f_{kj},\ f_{kj} + \Delta f_{kj} \in [0,\ 1] \qquad Q \rightarrow \begin{array}{l} \forall\Delta q_{ik},\ q_{ik} + \Delta q_{ik} \in [0,\ 1] \\ \forall\Delta q_{ik},\ \sum_k^K \Delta q_{ik} = 0 \end{array} \quad \because \sum_k^K q_{ik} = 1$$

# Ohana—Optimization

## Derivatives of the objective $P_1^O$ with respect to admixture proportions

$$\frac{\partial \left(\ln \left[P_1^O\left(Q,\ F\right)\right]\right)}{\partial q_{ik}} = \sum_j^J \left[\frac{g_{ij} \cdot f_{kj}}{\sum_m^K q_{im} \cdot f_{mj}} + \frac{(2 - g_{ij}) \cdot (1 - f_{kj})}{\sum_m^K q_{im} \cdot (1 - f_{mj})}\right]$$

$$\frac{\partial^2 \left(\ln \left[P_1^O\left(Q,\ F\right)\right]\right)}{\partial q_{ik} \partial q_{i'k'}} = \begin{cases} \sum_j^J \left\{ \frac{g_{ij} \cdot f_{kj} \cdot f_{k'j}}{\left(\sum_m^K q_{im} \cdot f_{mj}\right)^2} + \frac{(2 - g_{ij}) \cdot \left(1 - f_{kj}\right) \cdot \left(1 - f_{k'j}\right)}{\left[\sum_m^K q_{im} \cdot (1 - f_{mj})\right]^2} \right\} & \text{if } i = i' \\ 0 & \text{if } i = i' \end{cases}$$

## Derivatives of the objective $P_1^O$ with respect to allele frequencies

$$\frac{\partial \left(\ln \left[P_1^O\left(Q,\ F\right)\right]\right)}{\partial f_{kj}} = \sum_i^I \left[\frac{g_{ij} \cdot q_{ik}}{\sum_m^K q_{im} \cdot f_{mj}} - \frac{(2 - g_{ij}) \cdot q_{ik}}{\sum_m^K q_{im} \cdot (1 - f_{mj})}\right]$$

$$\frac{\partial^2 \left(\ln \left[P_1^O\left(Q,\ F\right)\right]\right)}{\partial f_{kj} \partial f_{k'j'}} = \begin{cases} \sum_j^J \left\{ \frac{g_{ij} \cdot q_{ik} \cdot q_{ik'}}{\left(\sum_m^K q_{im} \cdot f_{mj}\right)^2} + \frac{(2 - g_{ij}) \cdot q_{ik} \cdot q_{ik'}}{\left[\sum_m^K q_{im} \cdot (1 - f_{mj})\right]^2} \right\} & \text{if } j = j' \\ 0 & \text{if } j \neq j' \end{cases}$$

# Ohana—Optimization

Derivatives of the objective $P_1^L$ with respect to admixture proportions

$$\frac{\partial \left(\ln \left[P_1^L\left(Q,\ F\right)\right]\right)}{\partial q_{ik}} = \sum_j^J \left[\frac{G_Q\left(i,j,k\right)}{F\left(i,j\right)}\right]$$

$$\frac{\partial^2 \left(\ln \left[P_1^L\left(Q,\ F\right)\right]\right)}{\partial q_{ik}\partial q_{i'k'}} = \begin{cases} \sum_j^J \left[\frac{F(i,j)\cdot H_Q(i,j,k,k') - G_Q(i,j,k)\cdot G_Q(i,j,k')}{F^2(i,j)}\right] & \text{if } i = i' \\ 0 & \text{if } i \neq i' \end{cases}$$

Derivatives of the objective $P_1^L$ with respect to allele frequencies

$$\frac{\partial \left(\ln \left[P_1^L\left(Q,\ F\right)\right]\right)}{\partial f_{kj}} = \sum_i^I \left[\frac{G_F\left(i,j,k\right)}{F\left(i,j\right)}\right]$$

$$\frac{\partial^2 \left(\ln \left[P_1^L\left(Q,\ F\right)\right]\right)}{df_{kj}df_{k'j'}} = \begin{cases} \sum_i^I \left[\frac{F(i,j)\cdot H_F(i,j,k,k') - G_F(i,j,k)\cdot G_F(i,j,k')}{F^2(i,j)}\right] & \text{if } j = j' \\ 0 & \text{if } j \neq j' \end{cases}$$

# Ohana—Optimization with Block Structure

Rather than solving the full system $\Theta\left(I^2K^2 \cdot (I + 2IK) + J^2K^2 \cdot 2JK\right) = \Theta\left(J^3K^3\right)$

Example:

$$I = 3$$
$$J = 4$$
$$K = 2$$

$$H_Q = \begin{bmatrix} * & * & 0 & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 & 0 \\ 0 & 0 & * & * & 0 & 0 \\ 0 & 0 & * & * & 0 & 0 \\ 0 & 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & * & * \end{bmatrix} \qquad H_F = \begin{bmatrix} * & * & 0 & 0 & 0 & 0 & 0 & 0 \\ * & * & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & * & * & 0 & 0 & 0 & 0 \\ 0 & 0 & * & * & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & * & * & 0 & 0 \\ 0 & 0 & 0 & 0 & * & * & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & * & * \\ 0 & 0 & 0 & 0 & 0 & 0 & * & * \end{bmatrix}$$

$$D_Q = \begin{bmatrix} * & * & * & * & * & * \end{bmatrix} \qquad D_F = \begin{bmatrix} * & * & * & * & * & * & * & * \end{bmatrix}$$

... we handle a collection of small operations $\Theta\left(IK^2 \cdot (I + 2K) + JK^2 \cdot 2K\right) = \Theta\left(JK^3\right)$

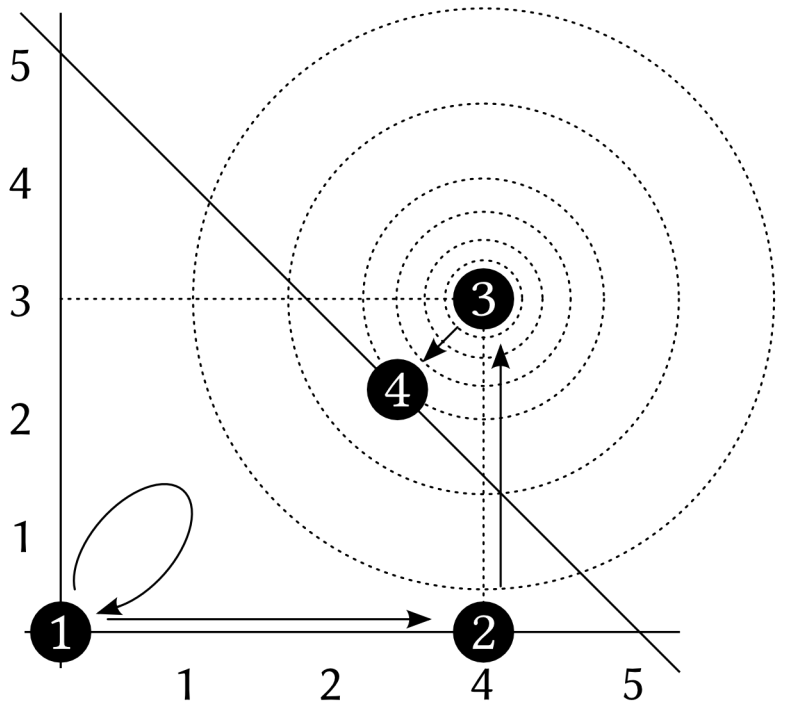
$$H_{Q_i} = \begin{bmatrix} * & * \\ * & * \end{bmatrix} \qquad H_{F_j} = \begin{bmatrix} * & * \\ * & * \end{bmatrix}$$

$$D_{Q_i} = \begin{bmatrix} * & * \end{bmatrix} \qquad D_{F_j} = \begin{bmatrix} * & * \end{bmatrix}$$

# Ohana—QP Active Set Algorithm

Concreate example:

$$\min_x \left\{ x^2 + y^2 - 8x - 6y \right\}$$
$$\text{s.t.} \; -x \leq 0$$
$$-y \leq 0$$
$$x + y \leq 5$$

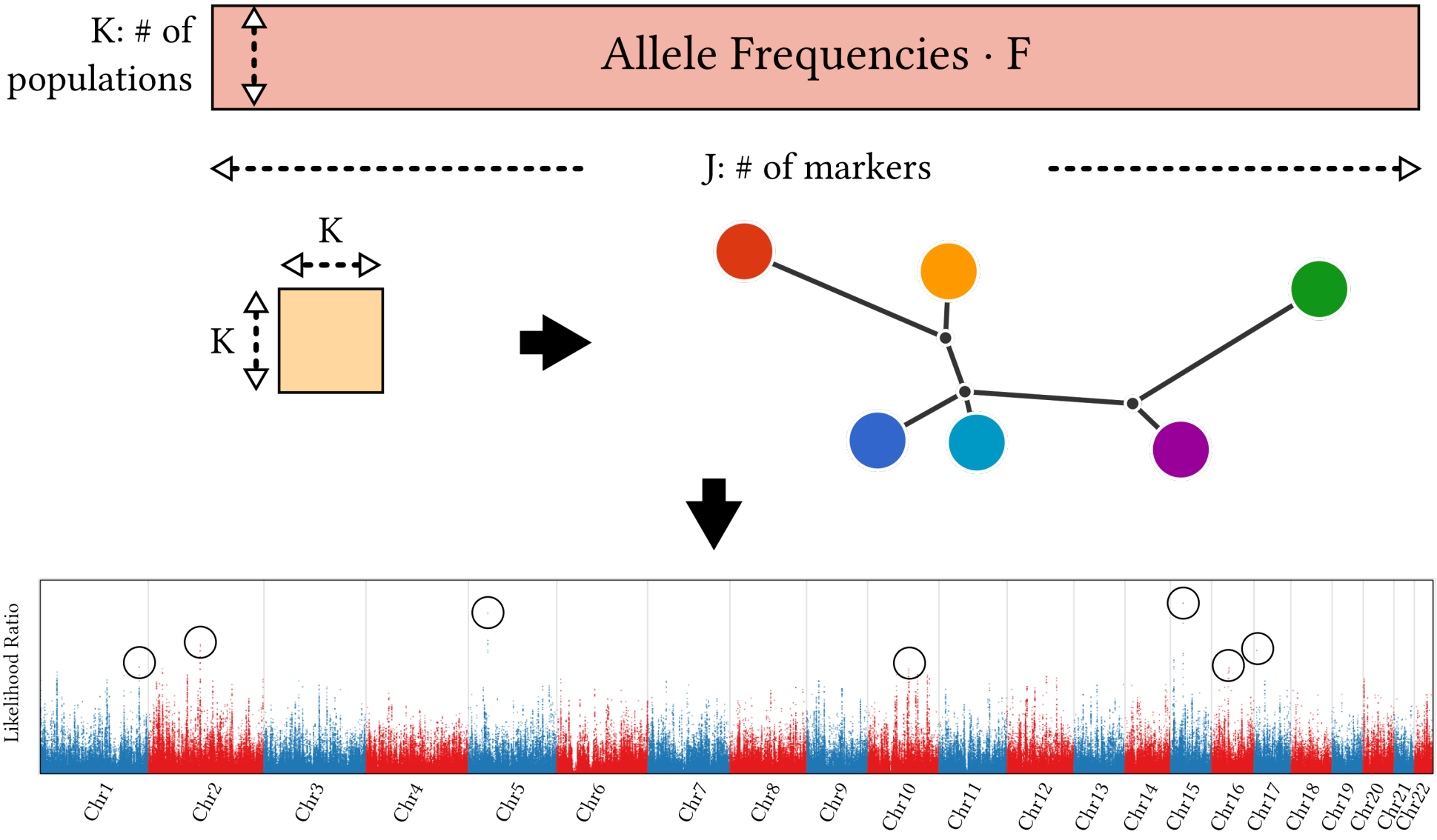



$$H = \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix}$$

$$D = \begin{bmatrix} -8 \\ -6 \end{bmatrix}$$

$$\min_x \left\{ \frac{1}{2} \begin{bmatrix} x & y \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} + \begin{bmatrix} -8 \\ -6 \end{bmatrix} \begin{bmatrix} x & y \end{bmatrix} \right\}$$

# Introduction to Ohana—Tree and Selection



# Ohana—Gaussian approximation

## Joint distribution of allele frequencies as a multivariate Gaussian

$$P\left(f_j \mid \Omega,\ \mu_j\right) \sim \mathcal{N}\left(\mu_j,\ \mu_j\left(1-\mu_j\right)\Omega\right)$$

$$\ln\left[P_2\left(F\right)\right] = -\frac{1}{2} \cdot \sum_j^J \left\{ K \cdot \ln\left(2\pi c_j\right) + \ln\left[\det\left(\Omega\right)\right] + \frac{1}{c_j} \cdot \left(f_j - \mu_j\right)^T \Omega^{-1} \left(f_j - \mu_j\right) \right\}$$

## Root all populations at an arbitrarily chosen population

$$\ln\left[P_2\left(F\right)\right] = -\frac{1}{2} \cdot \sum_j^J \left\{ \left(K-1\right) \cdot \ln\left(2\pi c_j\right) + \ln\left[\det\left(\Omega'\right)\right] + \frac{1}{c_j} \cdot f_j'^T \cdot \Omega'^{-1} \cdot f_j' \right\}$$

where $c_j = \mu_j\left(1-\mu_j\right)$

$$f_j' = f_j - f_{j0}$$

# Ohana—Phylogenetic Tree Estimation

```
3 3
0.33 0.22 0.16
0.22 0.56 0.15
0.16 0.15 0.24
```

Step #1: the rooted covariance matrix estimated for 4 populations

```
4 4
0.00 0.00 0.00 0.00
0.00 0.33 0.22 0.16
0.00 0.22 0.56 0.15
0.00 0.16 0.15 0.24
```

Step #2: the full covariance matrix estimated for 4 populations

```
4
A 0.00 0.33 0.56 0.24
B 0.33 0.00 0.45 0.25
C 0.56 0.45 0.00 0.50
D 0.24 0.25 0.50 0.00
```

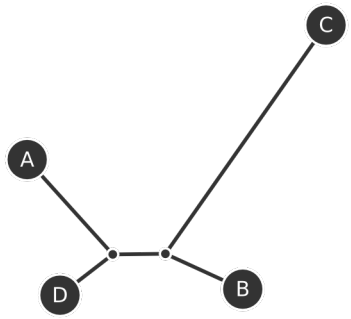
Step #3 the distance matrix in PHYLIP format, calculated from the full covariance matrix

convert cov2nwk

```
(A:0.157667,(C:0.345000,B:0.105000):0.065000,D:0.082333);
```

Step #4: the phylogenetic tree constructed from the distance matrix

convert nwk2svg
or the following service for even better quality graphics
`http://www.jade-cheng.com/graphs/`




Step #5: visualization of the phylogenetic tree

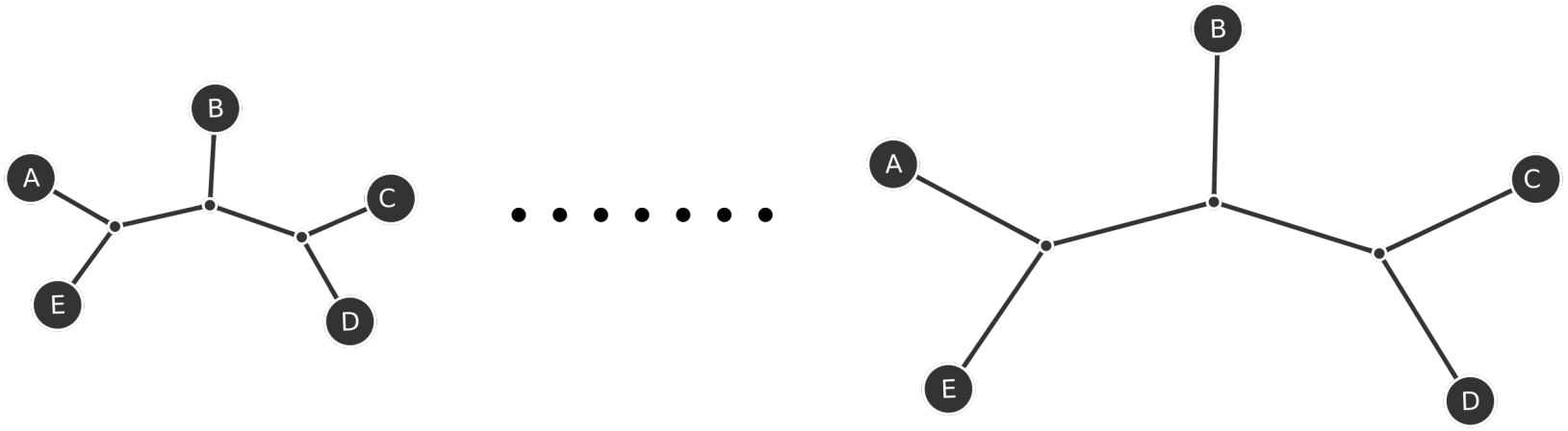
# Ohana—Selection

Selection increases the variance among populations.
Test each marker if it favors a larger variance.
Similar information as FST-based selection methods, but unsupervised.

$$P\left(f_j \mid \Omega,\ \mu_j\right) \sim \mathcal{N}\left(\mu_j,\ \mu_j\left(1-\mu_j\right)\Omega\right)$$

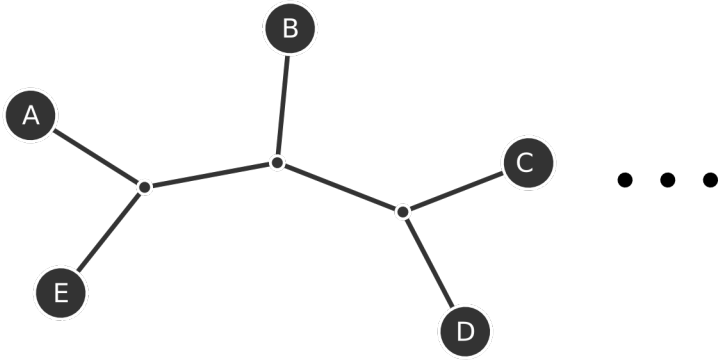
$$\ln\left[P_2\left(F\right)\right] = -\frac{1}{2}\cdot\sum_{j}^{J}\left\{K\cdot\ln\left(2\pi c_j\right)+\ln\left[\det\left(\Omega\right)\right]+\frac{1}{c_j}\cdot\left(f_j-\mu_j\right)^T\Omega^{-1}\left(f_j-\mu_j\right)\right\}$$

Example: scaling factor applied to the entire covariance structure.

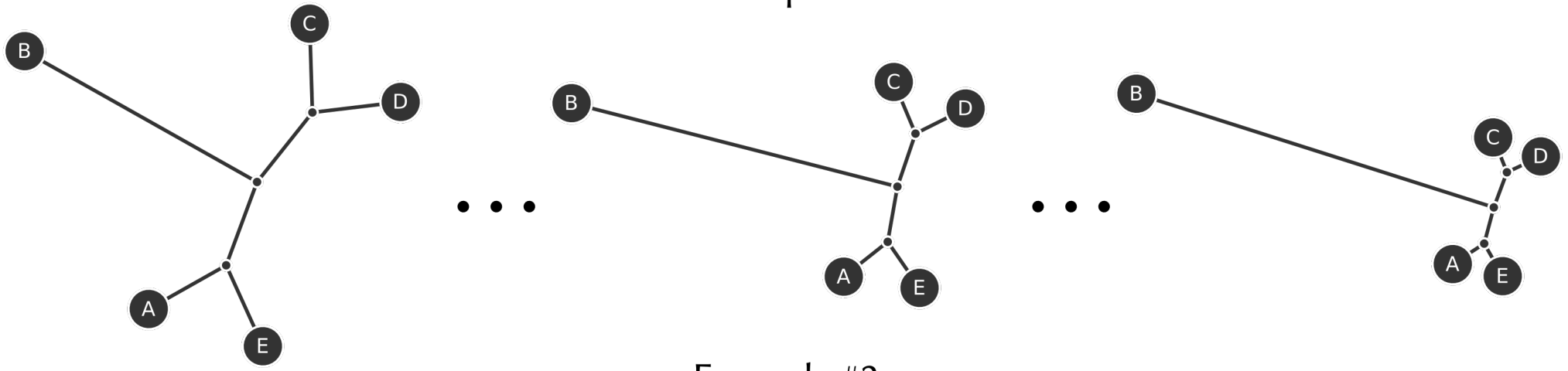


# Ohana—Selection

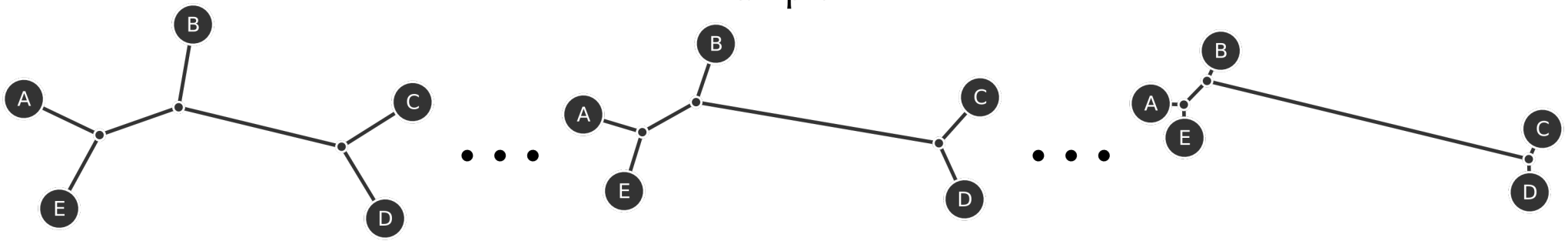
More examples: scaling factor applied to a portion of the covariance structure.



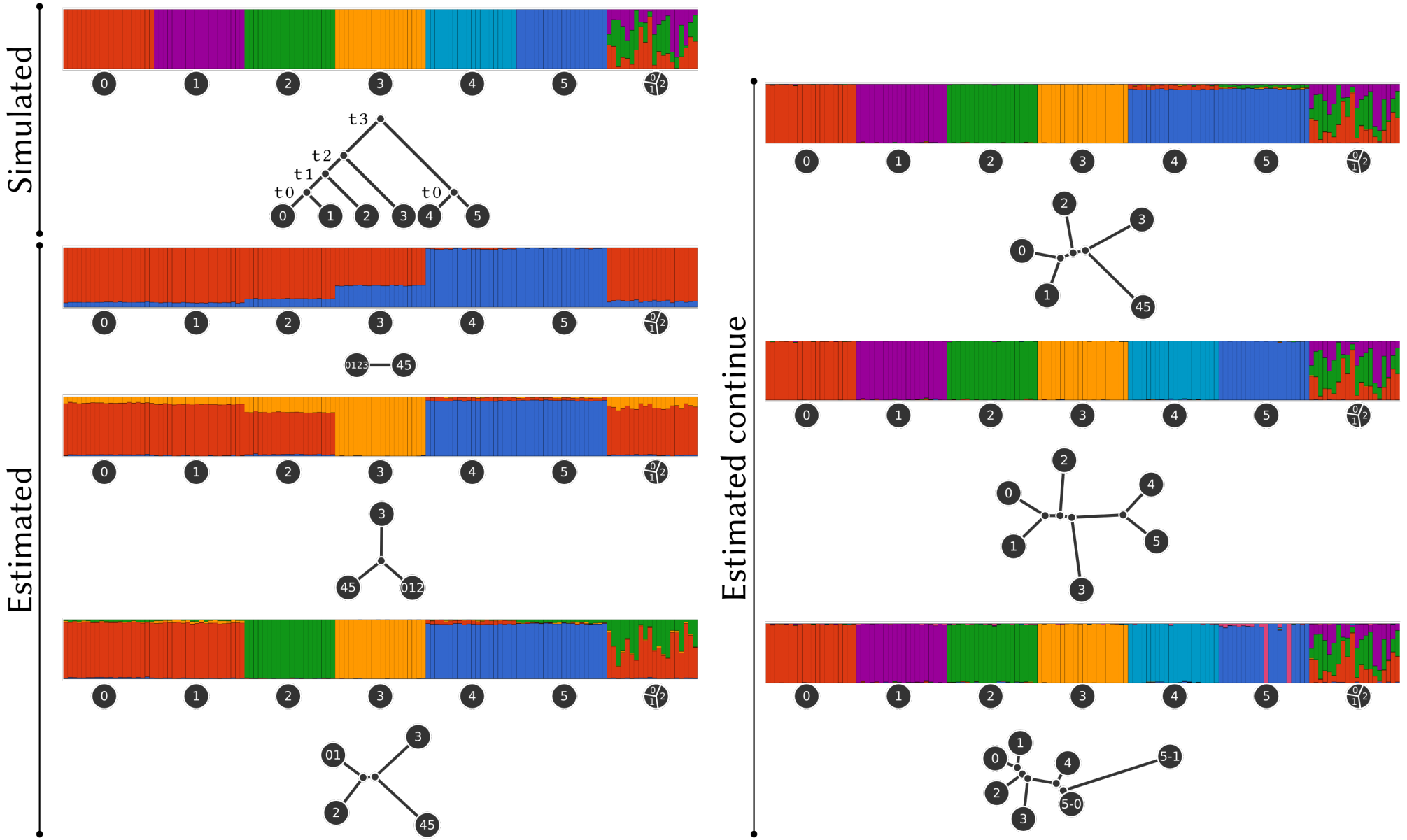

Example #1



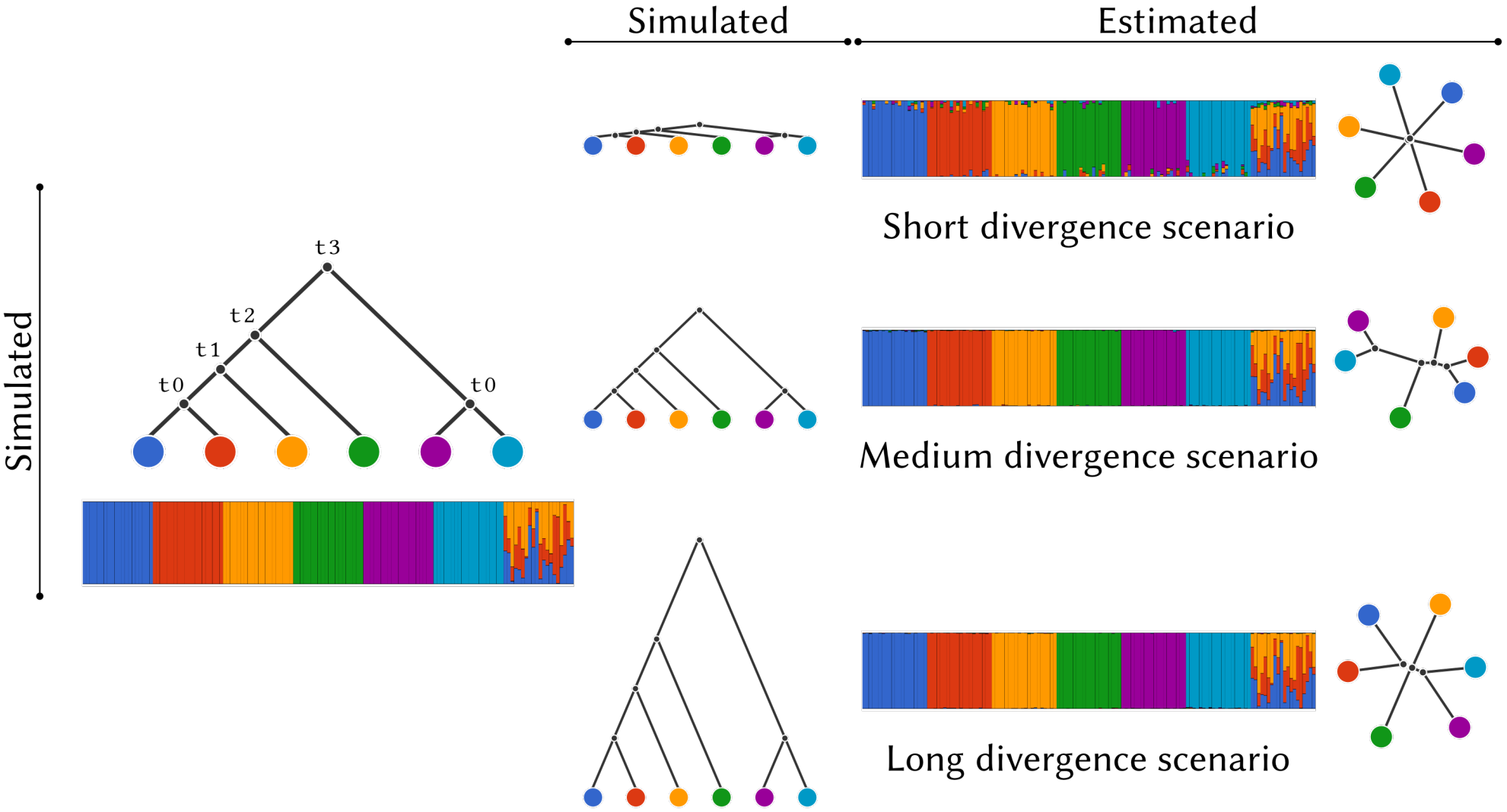

Example #2



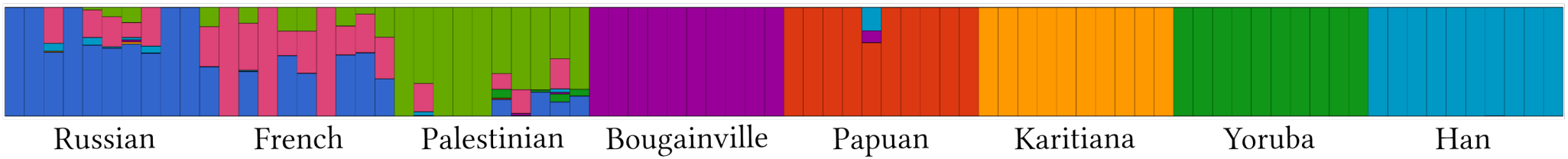
# Ohana—Simulation

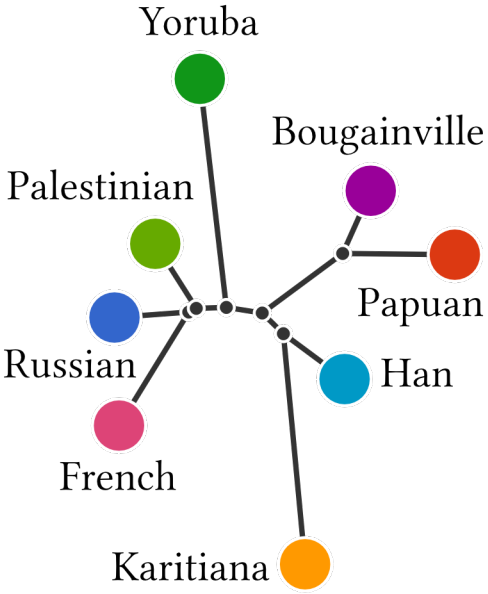


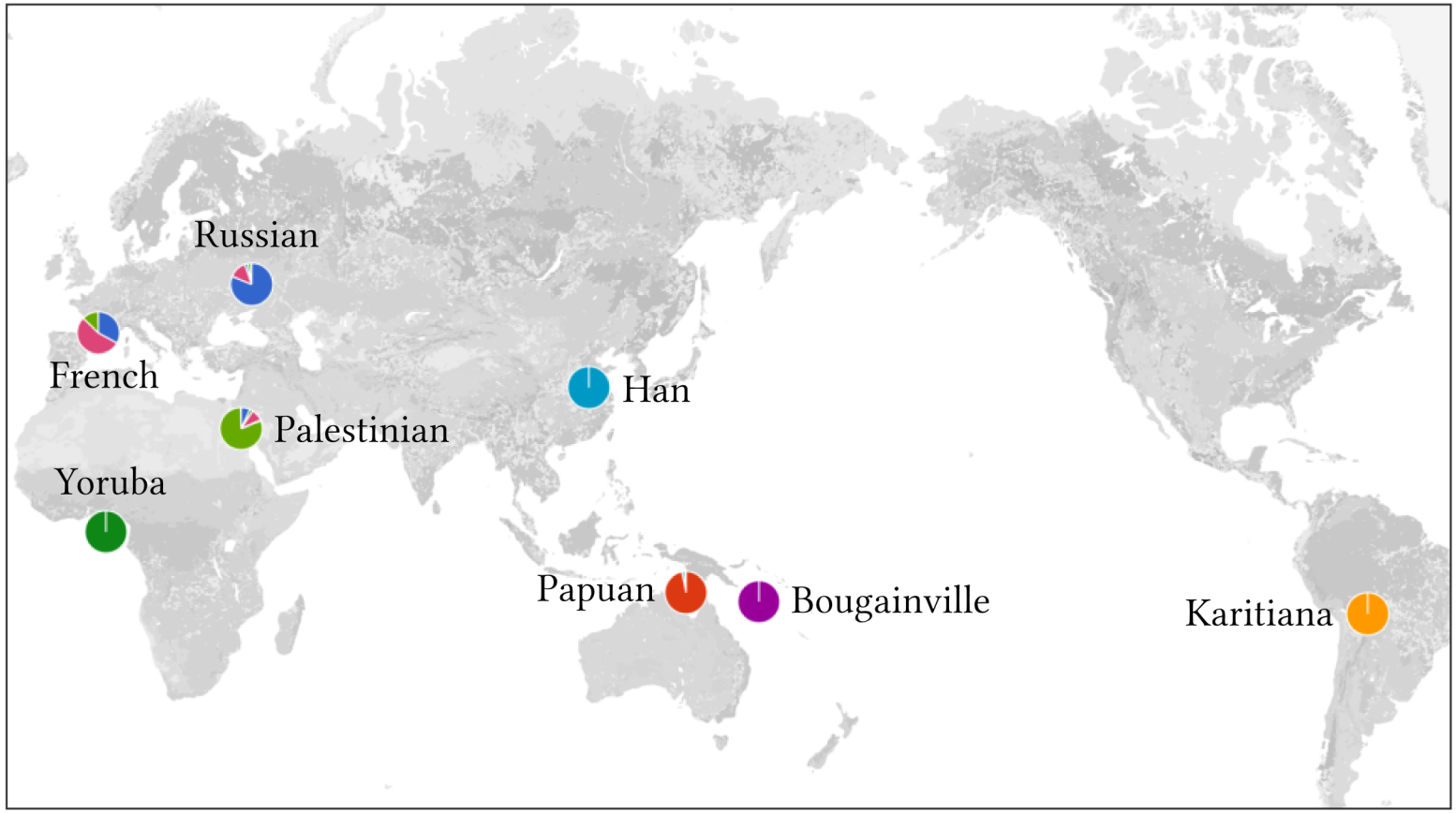
# Ohana—Different divergences



# Ohana—Biological data



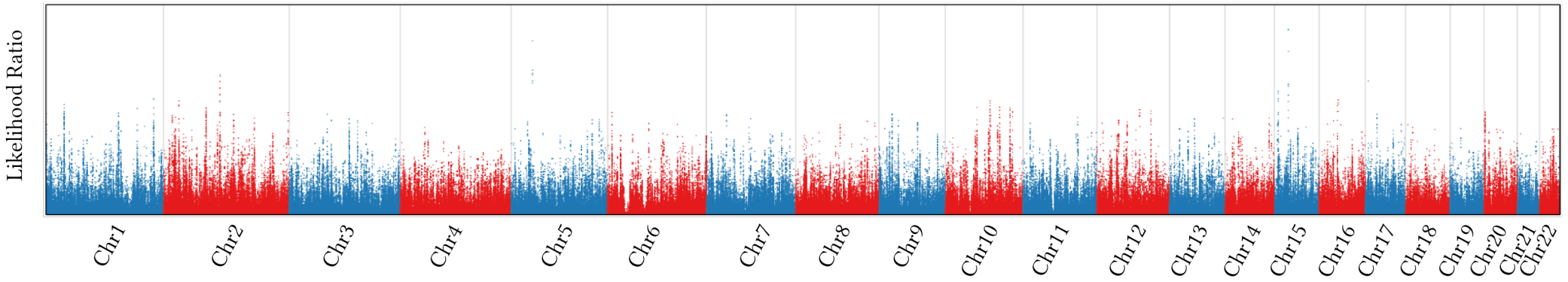



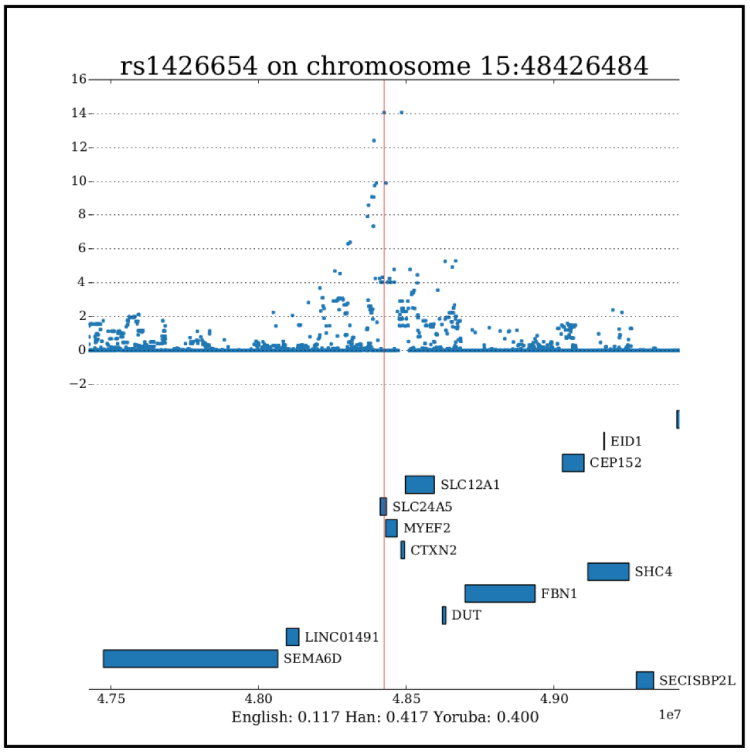



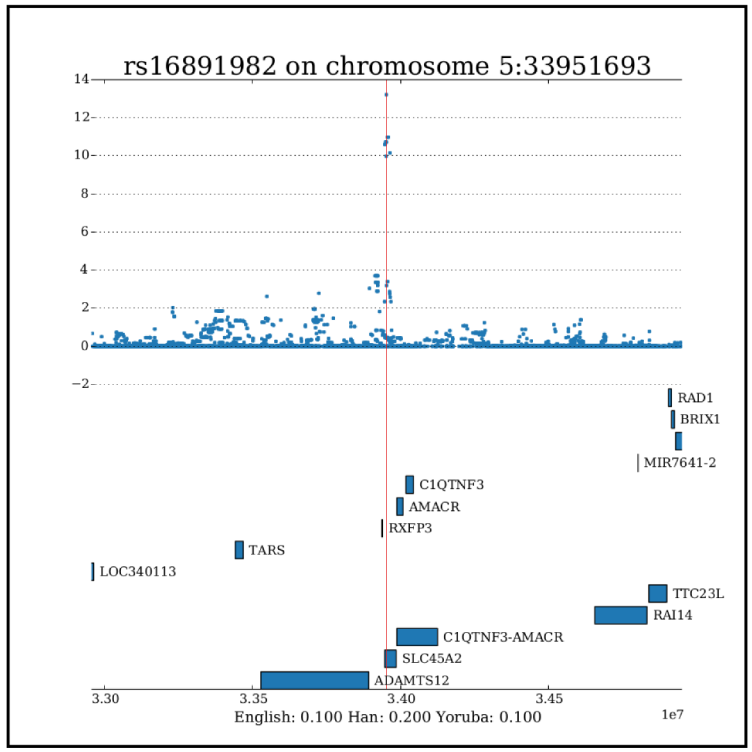

Map data © 2016 Google, INEGI · Phylogenetic trees: http://www.jade-cheng.com/graphs/

# Ohana—Selection from real data

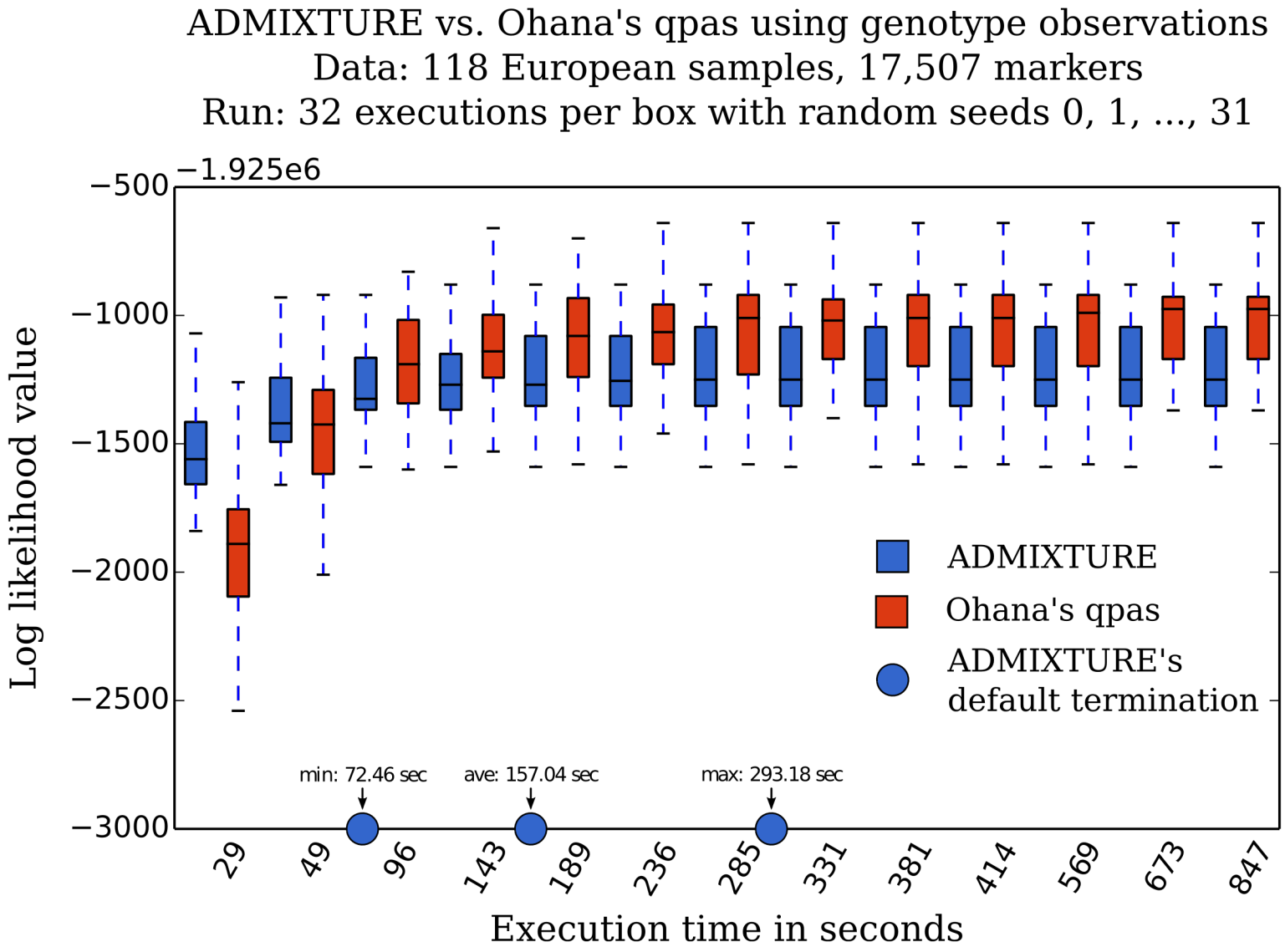
## A selection scan for English, Han, and Yoruba



# Ohana—Performance comparison



# Ohana—Performance comparison

Highest likelihood comparison

Each program each K, 100 executions using random seeds 0, 1, ..., 99

| | Dataset #1 | | | Dataset #2 | | | Dataset #3 | | |
|---|---|---|---|---|---|---|---|---|---|
| K | Ohana | ADMIXTURE | Diff | Ohana | ADMIXTURE | Diff | Ohana | ADMIXTURE | Diff |
| 2 | -1967733 | -1967733 | 0 | -3835358 | -3835365 | 7 | -1857263 | -1857263 | 0 |
| 3 | -1956785 | -1956799 | 14 | -3799873 | -3799887 | 14 | -1848450 | -1848451 | 1 |
| 4 | -1946218 | -1946244 | 26 | -3788598 | -3788607 | 10 | -1841198 | -1841199 | 1 |
| 5 | -1935775 | -1936025 | 250 | -3777351 | -3777361 | 11 | -1834377 | -1834378 | 1 |
| 6 | -1925636 | -1925877 | 241 | -3766558 | -3766540 | -18 | -1827829 | -1827830 | 2 |
| 7 | -1915552 | -1915743 | 191 | -3755851 | -3755860 | 9 | -1821445 | -1821458 | 13 |
| 8 | -1905430 | -1905638 | 209 | -3746227 | -3745412 | -815 | -1815214 | -1815214 | 0 |
| 9 | -1895372 | -1895879 | 507 | -3735240 | -3736079 | 839 | -1809084 | -1809101 | 18 |
| 10 | -1885306 | -1885466 | 160 | -3725558 | -3725624 | 66 | -1802911 | -1802906 | -5 |
| 11 | -1875503 | -1875853 | 350 | -3715543 | -3715157 | -385 | -1796763 | -1796847 | 84 |
| 12 | -1865492 | -1865965 | 474 | -3706069 | -3707715 | 1646 | -1790671 | -1790811 | 140 |
| 13 | -1855502 | -1856262 | 760 | -3697531 | -3698519 | 987 | -1784688 | -1784765 | 77 |
| 14 | -1845732 | -1846490 | 758 | -3688970 | -3689124 | 154 | -1778599 | -1778671 | 73 |
| 15 | -1836315 | -1836775 | 460 | -3681092 | -3680829 | -263 | -1772555 | -1772669 | 114 |



# Summary

I developed mathematical methods and software tools

| | |
|---|---|
| Admixture CoalHMM: | *https://github.com/jade-cheng/Jocx* |
| Ohana: | *https://github.com/jade-cheng/Ohana* |
| | *http://jade-cheng.com/ohana* |

... to study admixture

| | |
|---|---|
| Admixture CoalHMM: | infers demographic parameters |
| Ohana: | infers population structure |
| | infers evolutionary tree |
| | identifies selection signals |

... from genomic data.

| | |
|---|---|
| Admixture CoalHMM: | use full genome |
| | from a few individuals |
| Ohana: | use site-independent polymorphic data |
| | from many individuals |



# Future Work

CoalHMM:

Immediate:
- Stand-alone software package

Longer term:
- Automated model construction
- Model comparison

Ohana:

Immediate:
- Simulation validation for selection modules
- Joint modeling of ancient and modern data

Longer term:
- Joint inference with penalty and other likelihood functions
- Relax the tree-like assumption

# *Thank You!*

# 谢谢大家



# Ohana—Joint inference

Idea:

infer with additional information using Structure modeling
with or without additional parameters

Addition:

penalty, rewards, or additional likelihood functions, in general

Requirements:

second-order differentiable; respect the block structure

Approach:

incorporate new derivatives into SQP to solve Q and F
solve additional parameters with simple NO like NM
fixing each party while solving the other, switch, and iterate

Examples:

penalize admixing to prefer unadmixed estimates
modeling of the F with multivariate or simple Gaussian

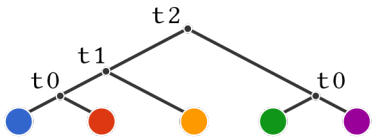


# Joint Modeling of Ancient and Modern Data

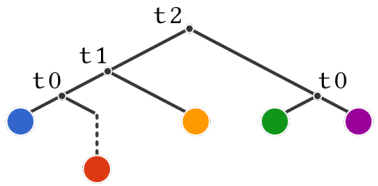
## Simulated

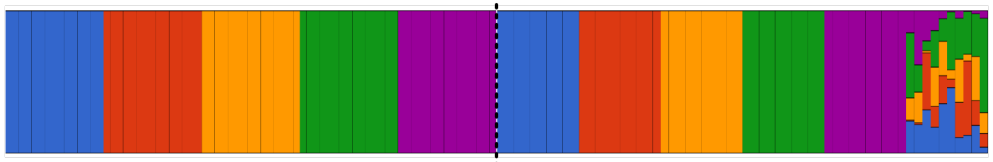
$$f_{\text{modern}} \sim \mathcal{N}\left(f_{\text{ancient}},\ f_{\text{ancient}}\left(1 - f_{\text{ancient}}\right)\sigma^2\right);\sigma^2 = 0.1$$

Ancient

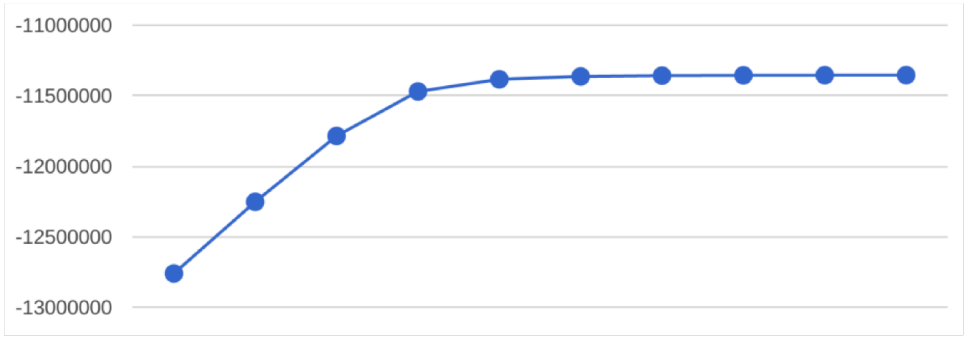



Modern



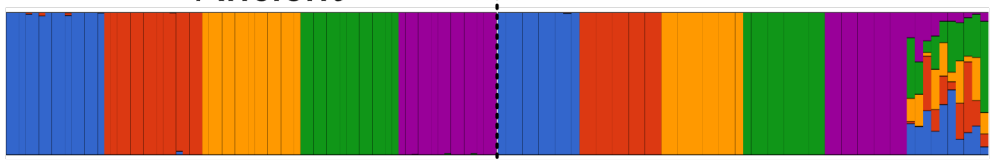
## Joint Log Likelihood



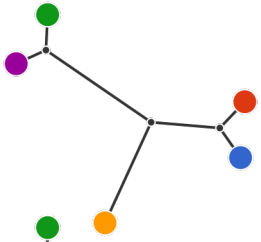

## Estimated

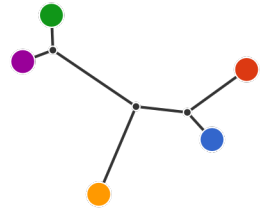
Ancient

Modern

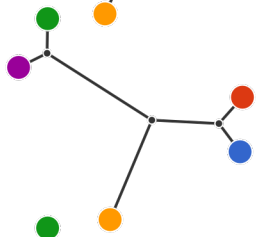


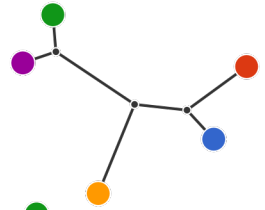
Iter #1



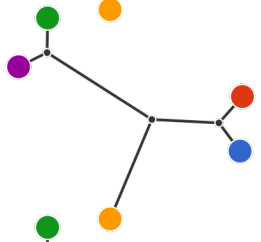


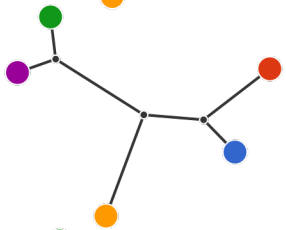
Iter #2





Iter #3





Iter #4