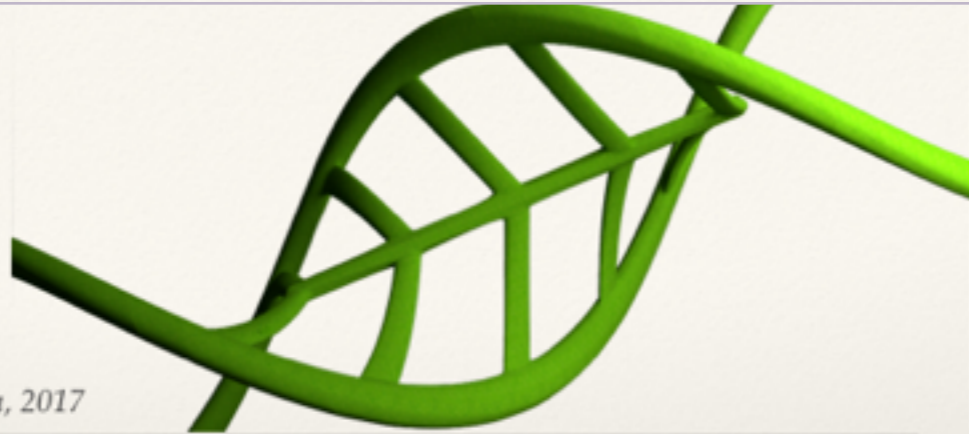


Montpellier, November 7th, 2017

Bioinformatic and analytical tools for the analysis of whole-genome sequence polymorphism data

Sebastián E. Ramos-Onsins

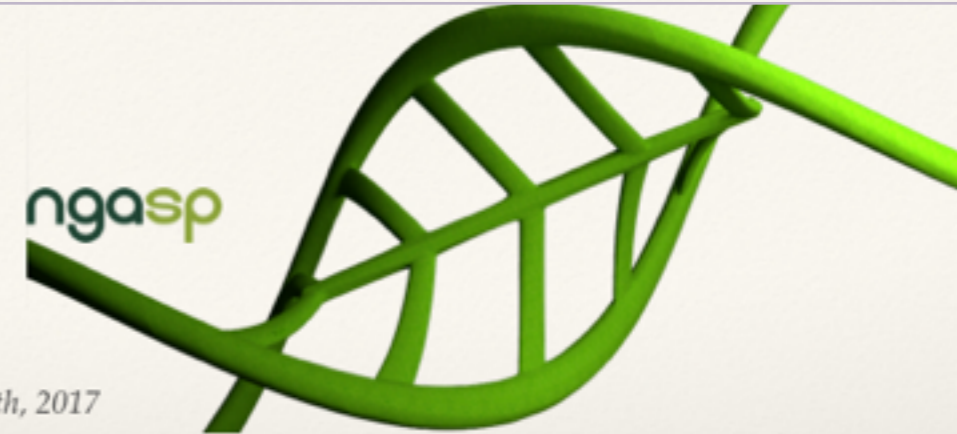




Montpellier, November 7th, 2017

DIGUP: Detection of Incompatible Genealogies Using Unphased Data

Mireia Vidal Villarejo
Luca Ferretti
Sebastián E. Ramos-Onsins



Montpellier, November 7th, 2017

ngasp: A Computational Tool for Population Genomic Analyses of NGS Datasets

Sebastián E. Ramos-Onsins
Gonzalo Vera





ngasp

Montpellier, November 7th, 2017

ngasp : A Computational Tool for Population Genomic Analyses of NGS Datasets

Sebastián E. Ramos-Onsins
Gonzalo Vera





- ❖ The *ngasp* (next generation analyses of sequence polymorphisms) starts from the necessity of having a user-friendly tool to perform the analysis of sequence variability dealing with NGS data.

ngasp



USER



DATA FILES:

- DNA SEQUENCES.
- FUNCTIONAL ANNOTATION.
- SPECIFIC REGIONS TO STUDY.

Load in
ngaSP

DETERMINE THE TYPE OF ANALYSIS:

- TYPE OF DATA.
- ORGANIZATION AND SELECTION OF DATA.
- FILTER DATA.
- SELECTION OF THE TYPE OF ANALYSIS.

PRE-ANALYSIS FOR RAW DATA:

- READ DEPTH AND SNP DETECTION

CALCULATION OF STATISTICS FOR OBSERVED DATA:

- ANALYSIS BY WINDOWS. *
- RESULTS IN TABLE AND PLOT FORMATS.

* Common code modules.

STATISTICAL INFERENCE:

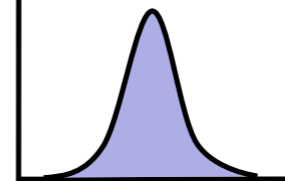
- COALESCENT SIMULATIONS.
- PERMUTATION TESTS.
- CALCULATION OF STATISTICS AND P-VALUES. *
- RESULTS IN TABLE AND PLOT FORMATS.

Output

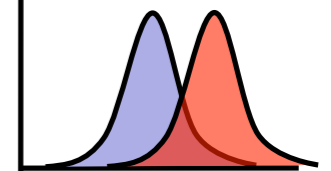
Output

RESULTS

SEQ1 12.23 ...
PSD2 45.21
...



SEQ1 12.23 Pvalue
PSD2 45.21 Pvalue
...





- ❖ Project in collaboration with computer engineers (in CRAAG and in the school of Engineers at the UAB)
- ❖ Includes multiple software tools for population genetic analysis (own-in-house and external):
 - ❖ manage BAM and gVCF and fasta files. Defined tfasta format.
 - ❖ SNP callers (pooled data, polyploid, diploid, haploid data).
 - ❖ Format converter tools.
 - ❖ Filter tools (BED files, GFF annotation).
 - ❖ Tools for sequence analyses with missing data.
 - ❖ Whole data or Sliding windows analysis..
 - ❖ Outputs: plots and / or tables.
- ❖ A web and graphical interface to manage project analysis as well as command line (JavaScript).
- ❖ Different kinds of users:
 - ❖ Experimental designer (final user)
 - ❖ Pipeline designer
 - ❖ Calculation designer
- ❖ Incorporating computational optimizations using distributed architectures.

Software for Analysis of Variability of NGS data



<https://bioinformatics.cragenomica.es/projects/ngaSP>

<https://github.com/cragenomica>

A screenshot of a web browser showing the GitHub organization page for CRAG. The browser's address bar shows the URL 'https://github.com/cragenomica'. The page header includes the GitHub logo, the organization name 'CRAG', and navigation links for 'This organization', 'Search', 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The main content area features the CRAG logo and the text 'CRAG - Centre for Research in Agricultural Genomics'. Below this, there are statistics for 'Repositories 22', 'People 4', 'Teams 2', and 'Projects 0'. A search bar for repositories is present, along with filters for 'Type: All' and 'Language: All', and a 'New' button. The repository list includes 'PFcaller' (Private, C, Updated 21 hours ago), 'indexingtFasta' (Private, C, Updated 3 days ago), 'gVCF2tFasta' (C, Updated 4 days ago), and 'lengthChromtFa' (C, Updated 4 days ago). A 'Top languages' section shows C, Shell, C++, Makefile, and Roff. A 'People' section shows 4 members with their profile icons.



<https://bioinformatics.cragenomica.es/projects/ngaSP>

The screenshot shows the ngasp website interface. At the top, there is a navigation bar with the ngasp logo and links for HOME, SOFTWARE, DOCS, and ABOUT. Below this is a large green banner with the ngasp logo and the tagline "next generation analysis of sequence polymorphisms". The main content area features a heading "Computational solution for performing next generation analysis of sequence polymorphisms using NGS data." followed by two paragraphs of text. To the right, a terminal window displays the ngasp website and terminal commands. A green callout box highlights the terminal output, with a sub-header "COMPUTATIONAL SOLUTION FOR PERFORMING NEXT GENERATION ANALYSIS OF SEQUENCE POLYMORPHISMS". The terminal output shows three commands and their results. At the bottom, there are two buttons: "HOW IT WORKS" and "GET THE SOFTWARE".

ngasp

HOME SOFTWARE DOCS ABOUT

ngasp

next generation analysis of sequence polymorphisms

Computational solution for performing next generation analysis of sequence polymorphisms using NGS data.

ngasp has been designed to calculate statistics analysis related to genome variability from NGS input data like genomes or exomes of individuals or even pooled data of population subsets. It will provide a series of analyses of importance to animal geneticists like tests to detect evidence of selection, differentiation, etc. It is foreseen that, in the future, can also accommodate phenotype data as soon as new analysis are developed and incorporated to ngasp.

This software is conceived to be used by different end-users, not only by specialists in the field but also by researchers interested in more common analyses (e.g., estimating variability). Other participants of this project, with user profiles like statisticians, tool developers or performance engineers are also better integrated easing the methods used to incorporate their contributions. As a result, ngasp will be able to read and represent graphically multiple input data formats, calculate a growing number of combined statistics, conveniently adjusted with a wide number of filters and options chosen by the user and output the results selecting between different tables and/or graphs, with varying degree of detail.

With the frontend

COMPUTATIONAL SOLUTION FOR PERFORMING NEXT GENERATION ANALYSIS OF SEQUENCE POLYMORPHISMS

With the backend

```
ngasp mstatspop -f fasta -i 100Kchr10.fa -o 1 -N 1 42 -T 100chr10.fa.txt
ngasp mstatspop -f ifa -i 100Kchr10.ifa -o 1 -N 5 20 20 2 -T 100chr10.ifa.txt
-G 1 -u 1 -w 100 -z 100 -f 1000 -s 1684
ngasp load -i script.ngasp
```

HOW IT WORKS GET THE SOFTWARE

Experimental designer



ngaSP localhost:3000/#

Experiment Editor Experiment 1

Experiment 1
Joan
This is the experiment for "pipeline 1".
End of line is OK!

Experiment Sessions List

Actions	Experiment Name	Experiment Progress
	→ Experiment 1	Finished

Experiment Results Console

```
Verbose Level: debug
00:00:00
> set-value --to $encoding --eq 'english_bn'
00:00:00
> print --text '[EXPERIMENT_START]' --eol
[EXPERIMENT_START]
00:00:00
> dim -n experiment_1/15_9_50/string_vector_2_0 --as string_vector
00:00:00
> set-value --to experiment_1/15_9_50/string_vector_2_0 --eq './examples/Banjo.chr12.20X.sorted.realigned.bam,./examples/Mini.chr12.20X.sorted.realigned.bam'
00:00:00
> dim -n experiment_1/15_9_50/string_3_0 --as string
00:00:00
> set-value --to experiment_1/15_9_50/string_3_0 --eq './examples/gorilla.chr12.fas'
00:00:00
> dim -n experiment_1/15_9_50/text-file_4_0 --as text-file
00:00:00
> set-value --to experiment_1/15_9_50/text-file_4_0 --eq './examples/gorilla.chr12.fas'
00:00:00
> dim -n experiment_1/15_9_50/string_1_0 --as string
00:00:00
> set-value --to experiment_1/15_9_50/string_1_0 --eq './output/statistics.txt'
00:00:00
> dim -n experiment_1/15_9_50/gtf-file_5_0 --as gtf-file
00:00:00
> set-value --to experiment_1/15_9_50/gtf-file_5_0 --eq './examples/file.gtf'
00:00:00
> dim -n experiment_1/15_9_50/bed-file_6_0 --as bed-file
00:00:00
> set-value --to experiment_1/15_9_50/bed-file_6_0 --eq './examples/file.bed'
00:00:00
> dim -n experiment_1/15_9_50/string_7_0 --as string
00:00:00
> set-value --to experiment_1/15_9_50/string_7_0 --eq '1 4'
00:00:00
```

Experiment Interactive Mode

Pipeline developer



ngaSP

127.0.0.1:3000/#

ngasp

Pipeline Editor

*pipeline_1 *pipeline_2 *pipeline_3A *pipeline_3B

pipeline_1
Sebas Ramos-Osins
Analysis of Variability of Haploid / Diploid samples from BAM files

```
graph LR; BAM[BAM Files] --> BAM_TO_MPLEUP[BAM TO MPLEUP]; Ref[Reference] --> BAM_TO_MPLEUP; BAM_TO_MPLEUP --> SNP_CALLER[SNP-CALLER]; SNP_CALLER --> CONCAT_FILES[CONCAT FILES]; Group[Outgroup] --> CONCAT_FILES; GTF[Annotation GTF File] --> CONCAT_FILES; Mask[Masking BED File] --> CONCAT_FILES; POP[Populations] --> CONCAT_FILES; Name[Output File Name] --> CONCAT_FILES; Word[File word] --> CONCAT_FILES; CONCAT_FILES --> BAMB_TO_BAMB[BAMB TO BAMB]; BAMB_TO_BAMB --> COLLECT_DATA_COLUMNS[COLLECT DATA COLUMNS]; COLLECT_DATA_COLUMNS --> Filtered[Filtered Statistics];
```

BAM TO MPLEUP

- BAM Files
- Reference
- Mpileup File
- Fasta Reference

SNP-CALLER

- Mpileup File
- Fasta File

CONCAT FILES

- Fast Input File
- Output File
- Sorted Input File

BAMB TO BAMB

- Fasta File
- Transposed Fasta File
- GTF File
- BED File

MISC/DIPOP

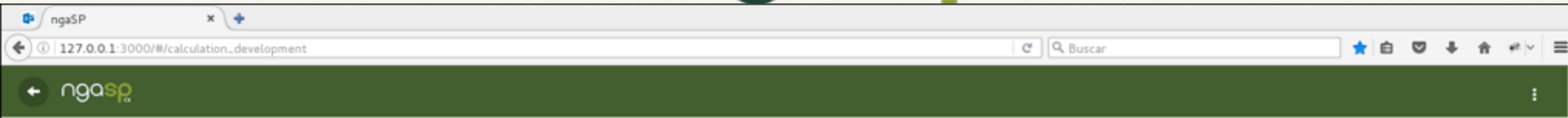
- File Format (f)
- Input File (i)
- Output Type (o)
- Populations (N)
- Outgroup Presence (G)
- Include Unknown (u)
- Output File Name (T)
- File HDF (a)
- File HDF (r)
- R2 Ploides (P)
- Sort mean (C)
- Miter (t)
- Seed (s)
- Window Size (w)
- Slide (z)
- Physical Length (Y)
- File word (W)
- File wps (E)
- Length (l)
- Miscdata (r)
- File Mask (m)
- Max svratio (v)
- Force Outgroup (F)
- Fileq revert (q)
- Ploidy (p)
- File GFF (g)
- Subset Positions (g)
- Code Name (g)
- Genetic Code (g)
- Criteria Transcript (c)
- Mask print (R)
- File Effz
- Statistics (S)
- Theta/nt(Wt)
- Theta/nt(Ta)
- Theta/nt(FuLU)
- Theta/nt(FaSW)
- Theta/nt(Deng)
- Theta/nt(Achaz,Wt)
- Theta/nt(Achaz,Ta)
- Divergence/nt
- Theta/nt(Taj,PKY)
- Divergence/PKY
- Theta/nt(Wt)
- Theta/nt(Ta)
- HajW
- rhap
- Tajima D
- FuLLI D
- FuLLI F
- FajSW norm H
- Deng E
- Achaz Y
- Fs
- SDev
- Skewness
- Kurtosis
- FstI
- FstH
- FstPKY
- FWPKY
- FWLBY

COLLECT DATA COLUMNS

- Required Column File
- Filtered Statistics File
- Statistics File

Filtered Statistics

Developer of Calculation boxes



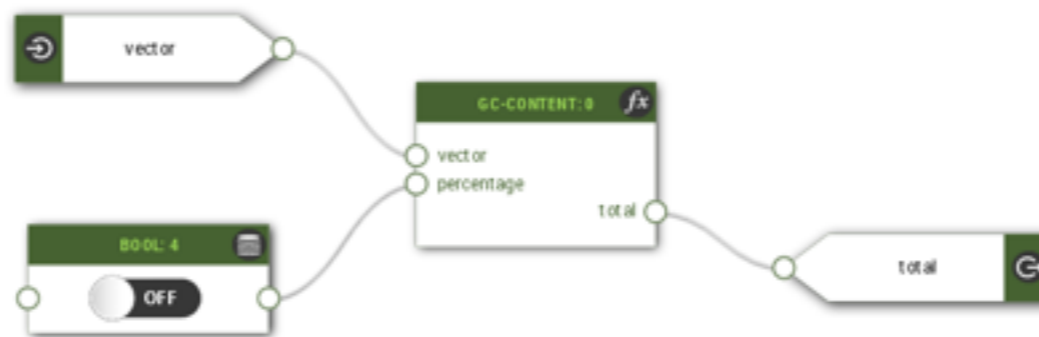
CALCULATION DEVELOPMENT

EXAMPLE. Creation of the "GC Content" calculation.

In molecular biology and genetics, GC-content (or guanine-cytosine content) is the percentage of nitrogenous bases on a DNA molecule that are either guanine or cytosine (from a possibility of four different ones, also including adenine and thymine).

GC content is usually expressed as a percentage value, but sometimes as a ratio (called G+C ratio or GC-ratio). GC-content percentage is calculated as $\frac{G + C}{A + T + G + C}$

whereas the AT/GC ratio is calculated as $\frac{A + T}{G + C}$



- 1 Create two constants for your calculation: one for the calculation's name and another for the calculation's description:

CSTRINGTABLE.H

```
enum KeyString {  
    ...  
    CALC_GCONTENT,  
    CALC_GCONTENT_DESC,  
    ...  
    _CALC_LAST,  
    ...  
}
```

- 2 Write your calculation name and description:

CSTRINGTABLE.CPP

```
CStringTable::CStringTable() {
```

Output results



ngaSP 127.0.0.1:3000/#

ngasp

*multiboxplot

Experiment Sessions List

Actions	Experiment Name	Experiment Progress
	→ multiboxplot	Finished

Experiment Results Console boxplot chart R Export

Iteration	Theta/nt(Wat)	Theta/nt(Ta)	Theta/nt(Fu&Li)
0	10.0	12.0	14.0
1	14.0	10.0	17.0
2	10.0	10.0	10.0
3	14.0	10.0	14.0
4	14.0	10.0	14.0
5	14.0	10.0	16.0
6	14.0	10.0	10.0
7	14.0	10.0	17.0
8	14.0	10.0	10.0
9	14.0	10.0	0.0

Iteration	Theta/nt(Wat)	Theta/nt(Ta)	Theta/nt(Fu&Li)
0	10.0	12.0	14.0
1	14.0	10.0	17.0
2	10.0	10.0	10.0
3	14.0	10.0	14.0
4	14.0	10.0	14.0
5	14.0	10.0	16.0
6	14.0	10.0	10.0
7	14.0	10.0	17.0
8	14.0	10.0	10.0
9	14.0	10.0	0.0

MEATPOP

- File Format (-f)
- Input File (-i)
- Output Type (-o)
- Populations (-M)
- Outgroup Presence (-G)
- Include Unknown (-u)
- Output File Name (-T)
- File H1F (-s)
- File H2F (-n)
- R3 Ploides (-P)
- Sort name (-O)
- Witer (-t)
- Seed (-e)
- Window Size (-w)
- Slide (-c)
- Physical Length (-Y)
- File vcoord (-W)
- File vgs (-E)
- Length (-l)
- SiteData (-r)
- File Mask (-m)
- Ms aevatio (-v)
- Force Outgroup (-F)
- Freq revert (-q)
- Ploidy (-p)
- File OFF (-g)
- Subset Positions (-g)
- Code Name (-g)
- Genetic Code (-g)
- Criteria Transcript (-c)
- Mask print (-K)
- File Effiz

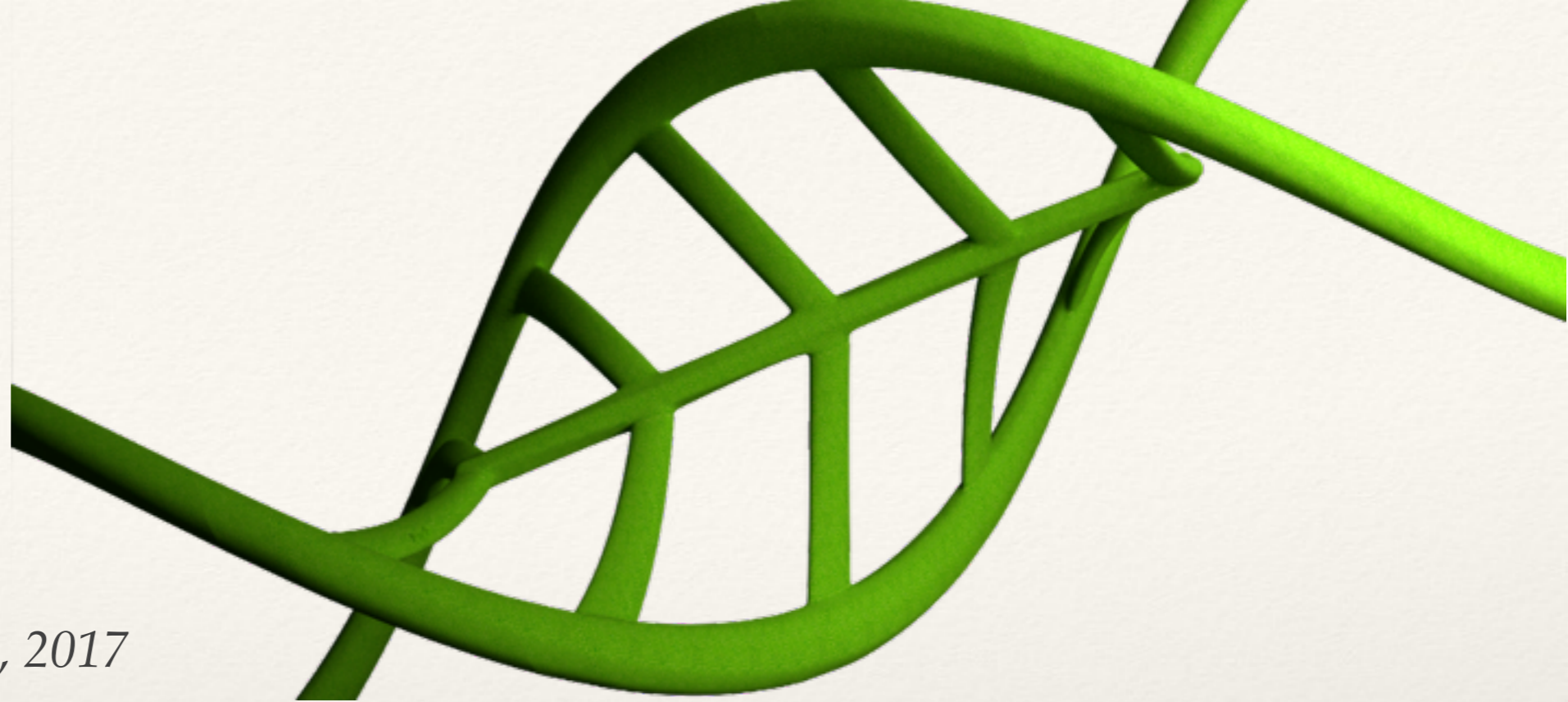
Statistics

- S
- Theta/nt(Wat)
- Theta/nt(Fu&Li)
- Theta/nt(Ta)
- Theta/nt(Achaz:Ta)
- Theta/nt(Achaz:Ta)
- Divergence/nt
- Theta/nt(Ta)HKY
- Divergence/HKY
- Theta/nt(Wat)
- Theta/nt(Ta)
- HapW
- nHap
- Tajima D
- Fu&Li D
- Fu&Li F
- Fay&Wu norm H
- Zeng E
- Achaz Y
- Fs
- SDev
- Skewness
- Kurtosis
- Fst
- FstH
- FstHKY
- PNNKY
- PNNKY

BOXPLOT

CHART

R EXPORT



Montpellier, November 7th, 2017

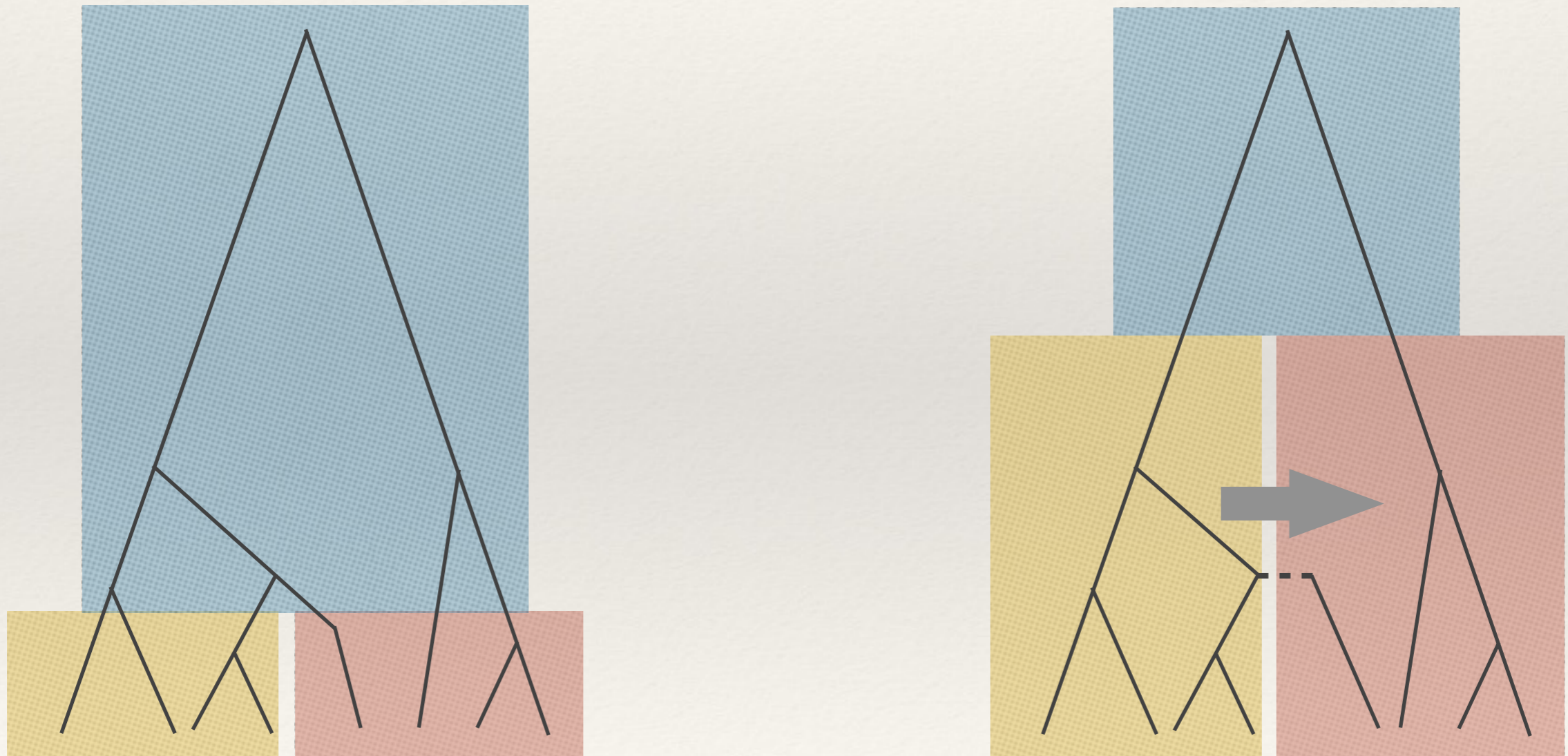
DIGUP: Detection of Incompatible Genealogies Using Unphased Data

Mireia Vidal Villarejo
Luca Ferretti
Sebastián E. Ramos-Onsins



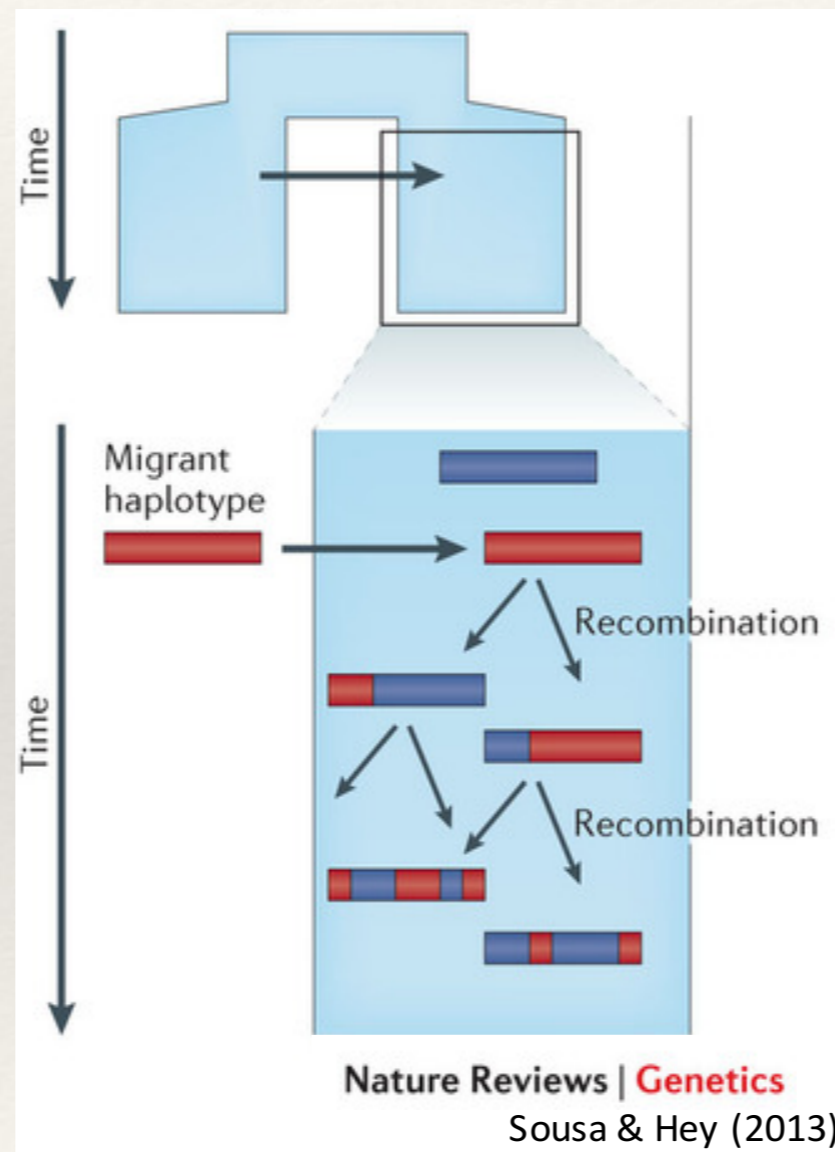
Ancestral Polymorphism vs Migration, Recombination and Incompatible Genealogies

- ❖ Ancestral Polymorphism and Migration can be confounded:



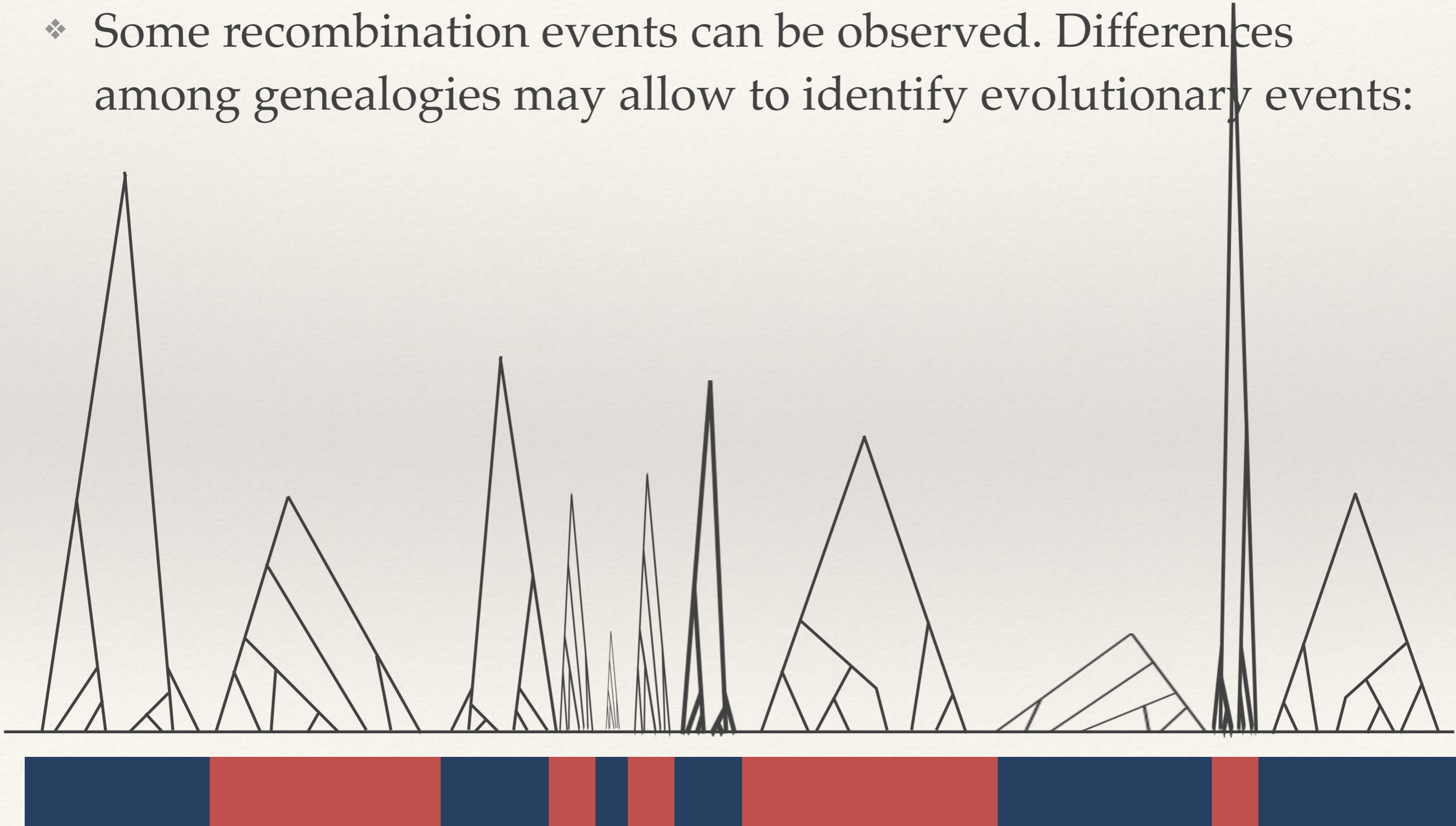
Ancestral Polymorphism vs Migration, Recombination and Incompatible Genealogies

- ❖ Recombination cut and join different genealogies:



Ancestral Polymorphism vs Migration, Recombination and Incompatible Genealogies

- ❖ Some recombination events can be observed. Differences among genealogies may allow to identify evolutionary events:



Ancestral Polymorphism vs Migration, Recombination and Incompatible Genealogies

- ❖ Pooled data adds complexity to the study:
 - ❖ For each position, different individuals are considered, and also different sample sizes can be used.
 - ❖ The genealogy of a region (or a position) can not be directly compared because the samples are different.
- ❖ Missing data can be considered as a similar problem, as we can have information from different individuals of the populations with different sample sizes per position.

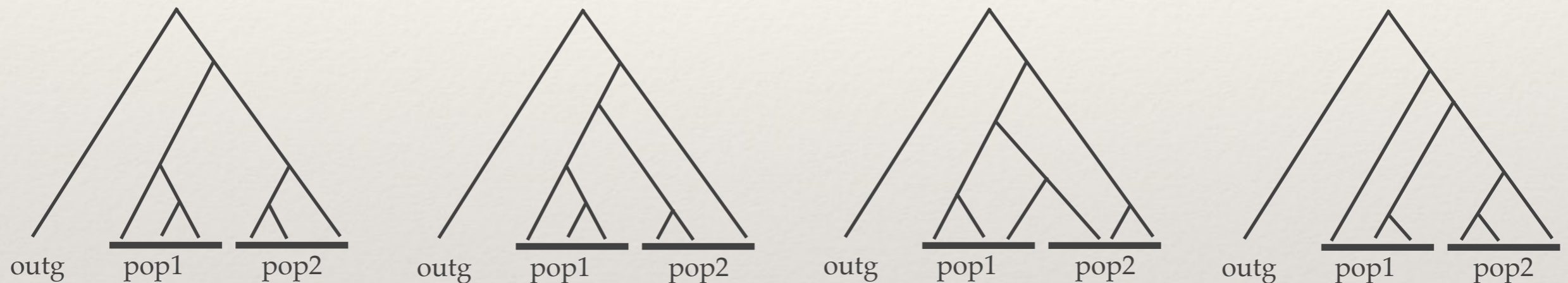
Study of the Variability in Populations



- ❖ We aim to:
 - ❖ Design simple statistics and algorithms that describe the variability among populations involved in the genome.
 - ❖ Detect incompatible genealogies and their lengths across the genome, using unphased data.
 - ❖ Study the expected patterns of these statistics (or algorithms) under different conditions.

Methodology

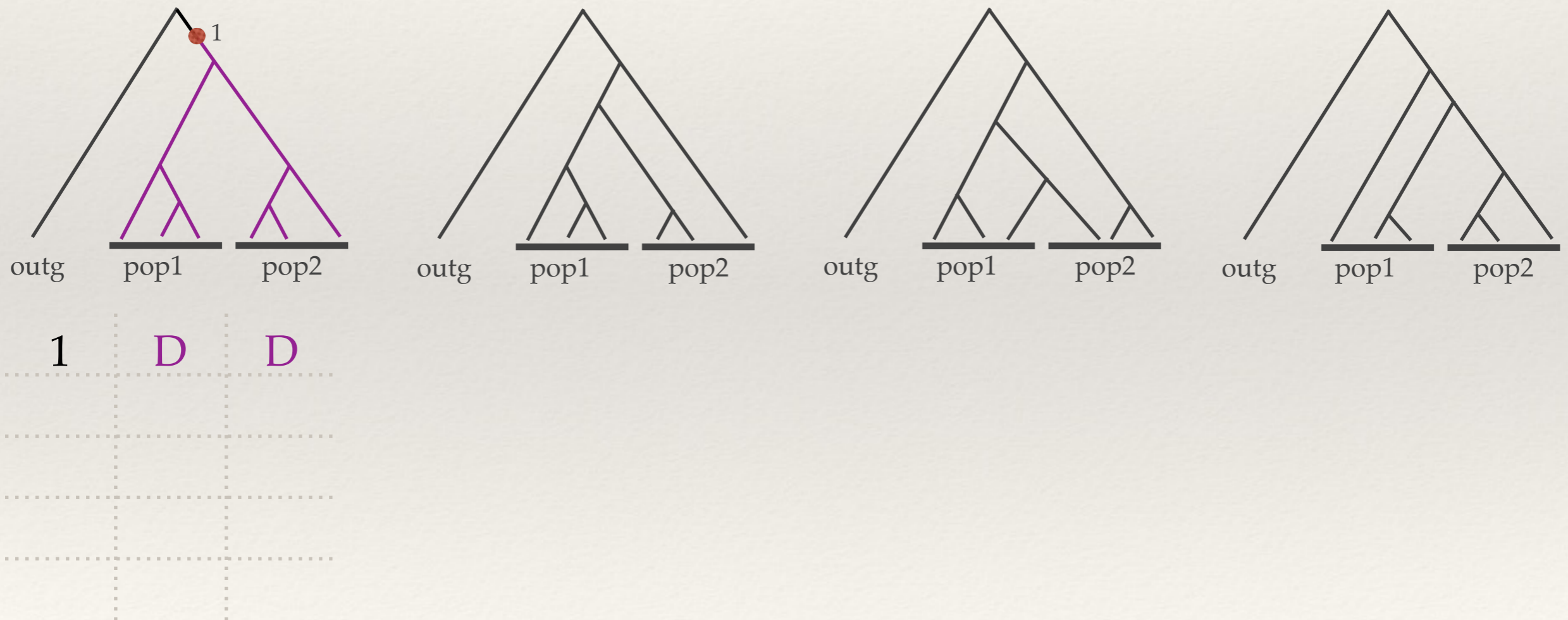
- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population (4 rooted genealogies):



- ❖ We define three possible states for each population:
 - ❖ **A**: Ancestral (all samples equal to the outgroup)
 - ❖ **D**: Derived (all samples different to the outgroup)
 - ❖ **P**: Polymorphic

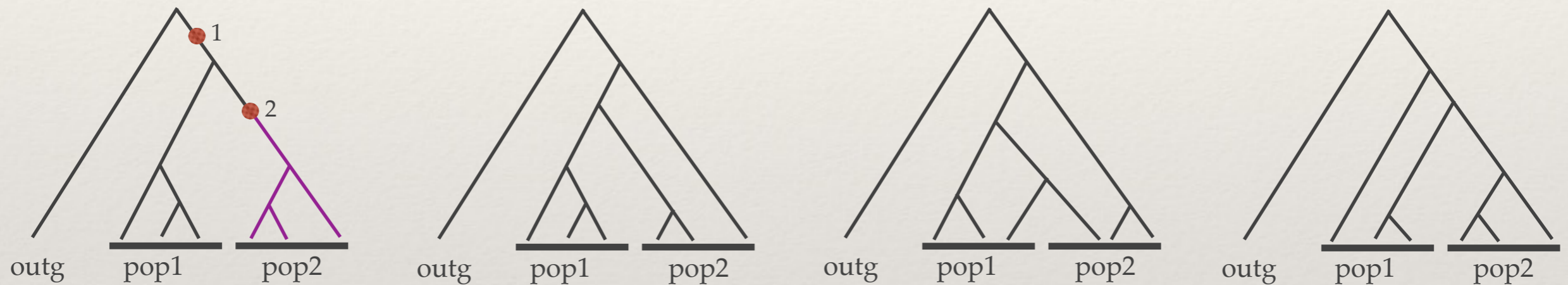
Methodology

- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:



Methodology

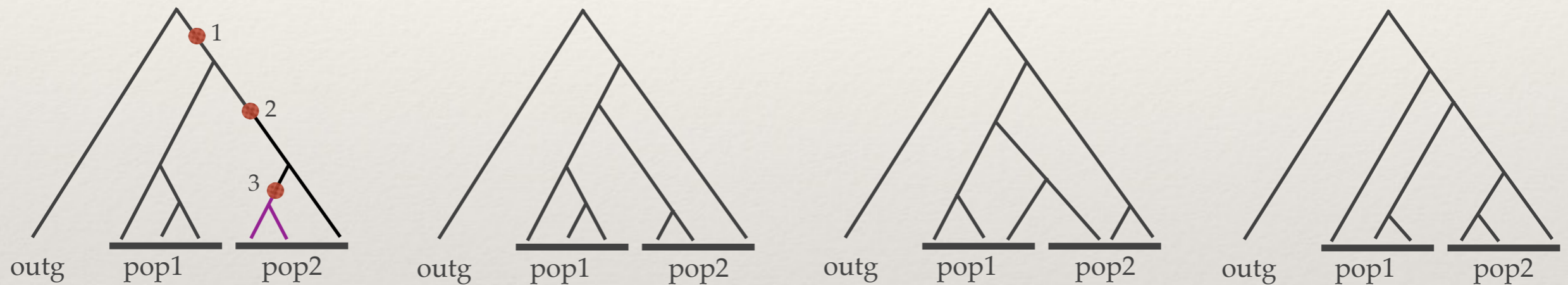
- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:



1	D	D
2	A	D

Methodology

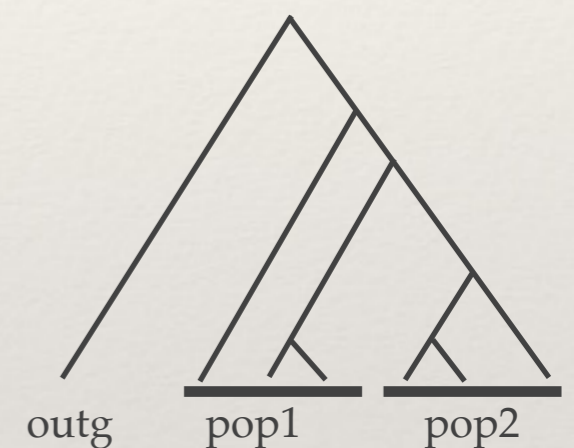
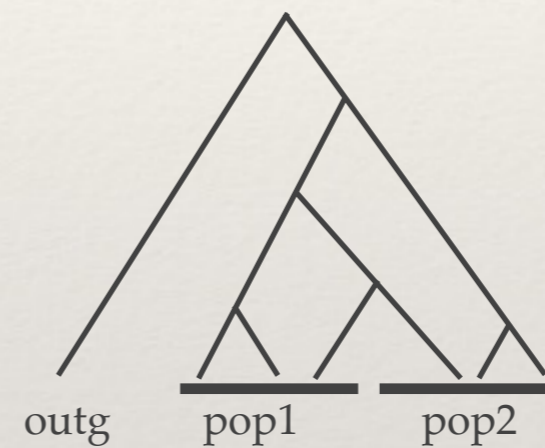
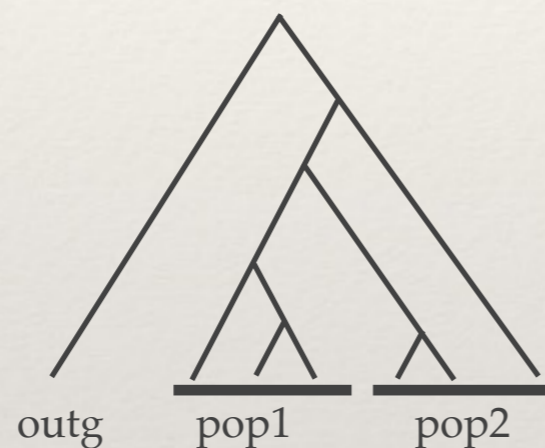
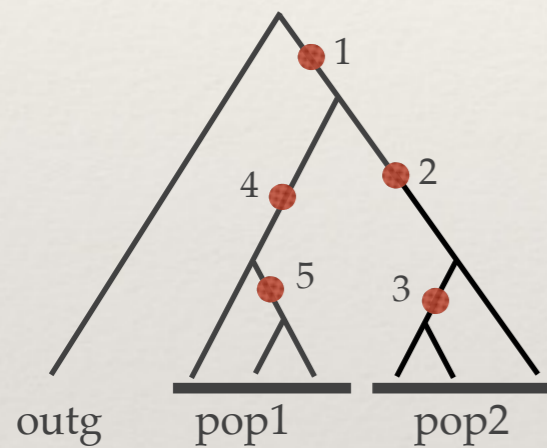
- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:



1	D	D
2	A	D
3	A	P

Methodology

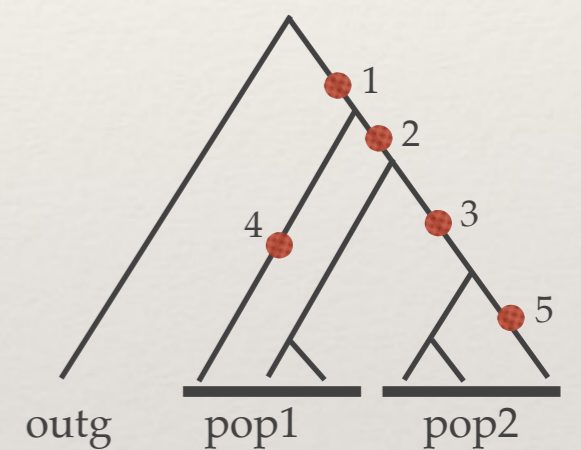
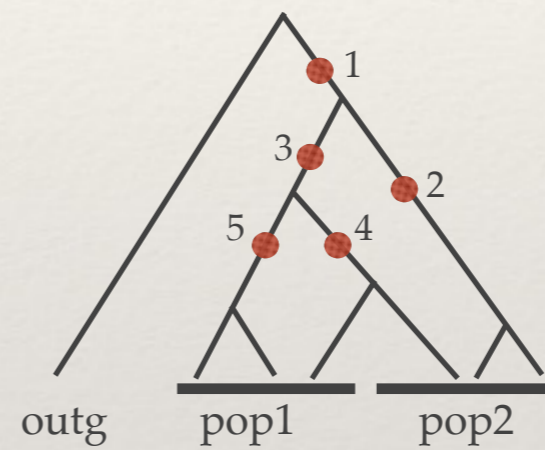
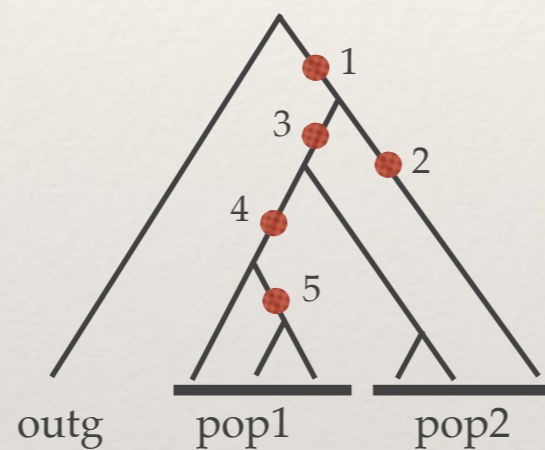
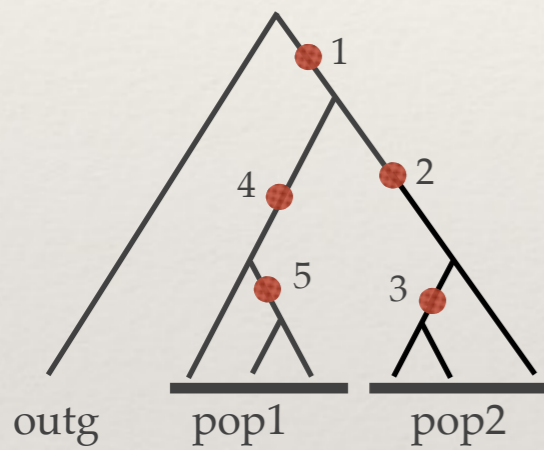
- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:



1	D	D
2	A	D
3	A	P
4	D	A
5	P	A

Methodology

- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:



1	D	D
2	A	D
3	A	P
4	D	A
5	P	A

1	D	D
2	A	P
3	D	P
4	D	A
5	P	A

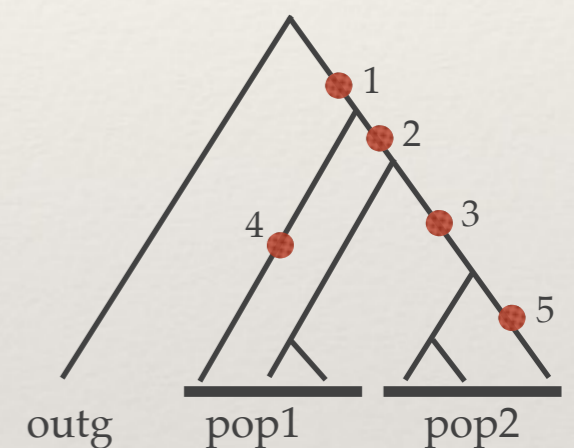
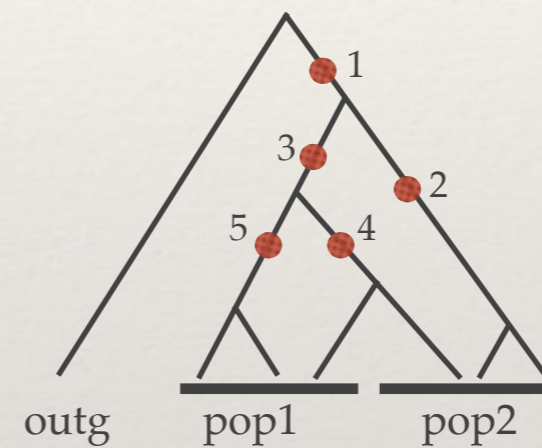
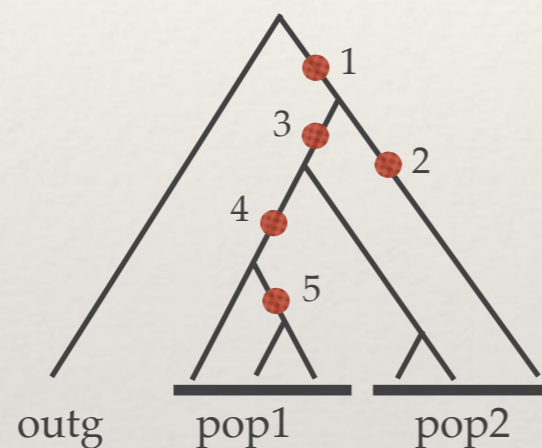
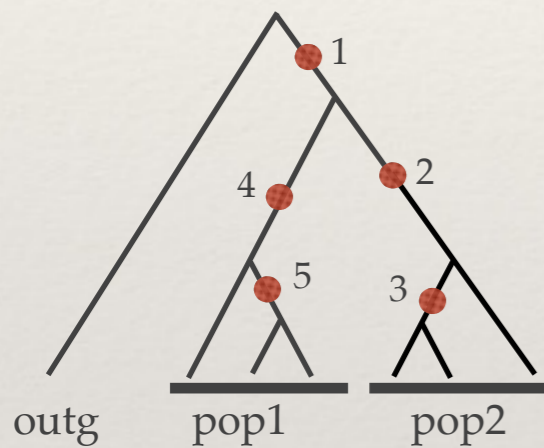
1	D	D
2	A	P
3	D	P
4	P	P
5	P	A

1	D	D
2	P	D
3	A	D
4	P	A
5	A	P

Methodology

- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:

COMPATIBLE COMBINATIONS?



1	D	D
2	A	D
3	A	P
4	D	A
5	P	A

1	D	D
2	A	P
3	D	P
4	D	A
5	P	A

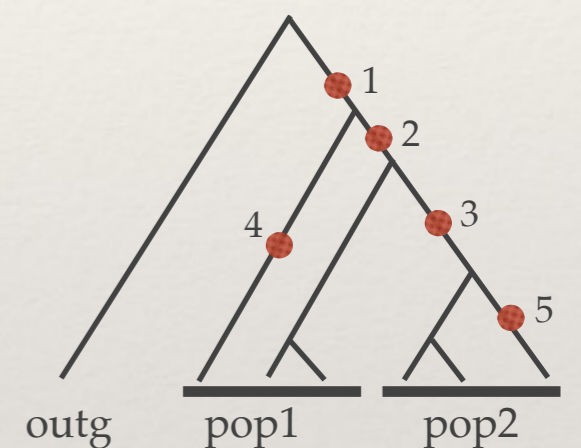
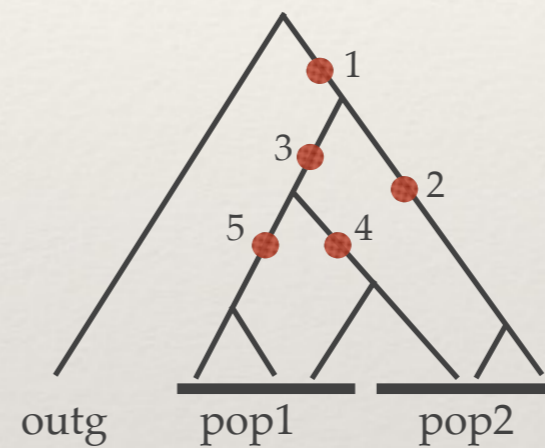
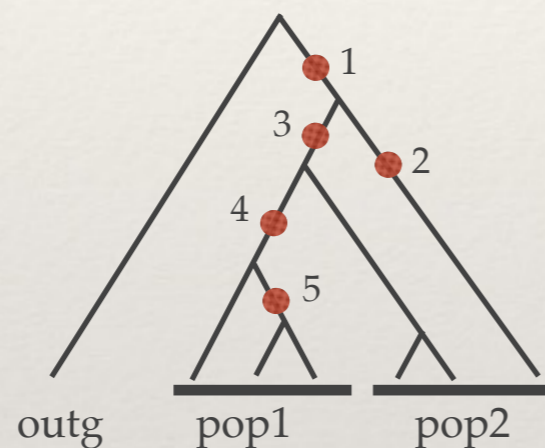
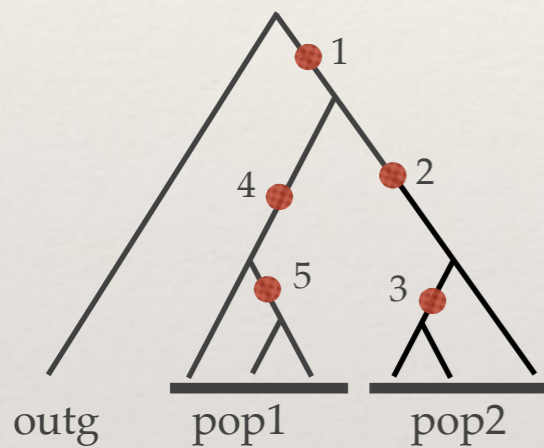
1	D	D
2	A	P
3	D	P
4	P	P
5	P	A

1	D	D
2	P	D
3	A	D
4	P	A
5	A	P

Methodology

- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:

INCOMPATIBLE COMBINATIONS?



1	D	D
2	A	D
3	A	P
4	D	A
5	P	A

1	D	D
2	A	P
3	D	P
4	D	A
5	P	A

1	D	D
2	A	P
3	D	P
4	P	P
5	P	A

1	D	D
2	P	D
3	A	D
4	P	A
5	A	P

Methodology

- ❖ Find incompatible genealogies along the genome considering TWO populations and one ancestral outgroup population:

INCOMPATIBLE COMBINATIONS

	AA	PA	AP	DD	PP	DA	AD	DP	PD
AA	AA								
PA		PA							
AP			AP						
DD				DD					
PP					PP				
DA						DA			
AD							AD		
DP								DP	
PD									PD

RED: incompatible combinations

Methodology

- ❖ Find incompatible genealogies along the genome considering THREE populations and one ancestral outgroup population (105 rooted bifurcating genealogies):

	AAA	AAP	APA	DDD	PAA	DDP	PPA	DAA	PAP	AAD	APP	ADA	DPD	PDD	DPP	PAD	DAP	APD	PDP	DDA	ADD	PPD	DPA	ADP	PDA	DAD	PPP		
AAA	AAA																												
AAP		AAP																											
APA			APA																										
DDD				DDD																									
PAA					PAA																								
DDP						DDP																							
PPA							PPA																						
DAA								DAA																					
PAP									PAP																				
AAD										AAD																			
APP											APP																		
ADA												ADA																	
DPD													DPD																
PDD														PDD															
DPP															DPP														
PAD																PAD													
DAP																	DAP												
APD																		APD											
PDP																			PDP										
DDA																				DDA									
ADD																					ADD								
PPD																						PPD							
DPA																							DPA						
ADP																								ADP					
PDA																									PDA				
DAD																										DAD			
PPP																												PPP	

RED: incompatible combinations in two populations.

GREEN: incompatible combinations in three populations.

Methodology

- ❖ From these two simple examples we infer two main rules of incompatibility:
 - ❖ Incompatibility between two pops:
 - AD vs DP**
 - AD vs PP**
 - PD vs DP**
 - ❖ Incompatibility between three pops:
 - DAX vs DXA** (X can be D or P)

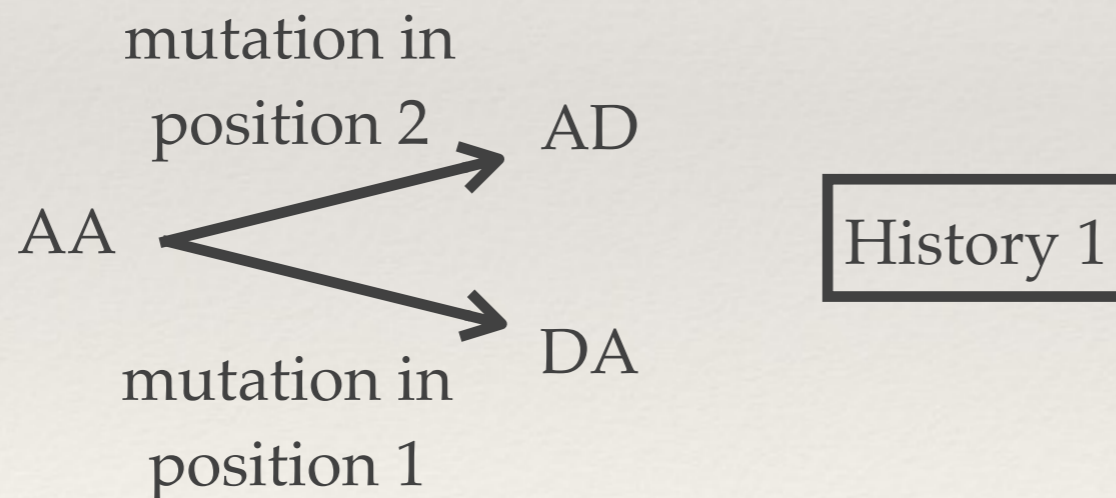
Methodology

- ❖ Find incompatible genealogies along the genome considering N populations and one ancestral outgroup:

All incompatibilities between states are obtained by combinations of two or three populations.

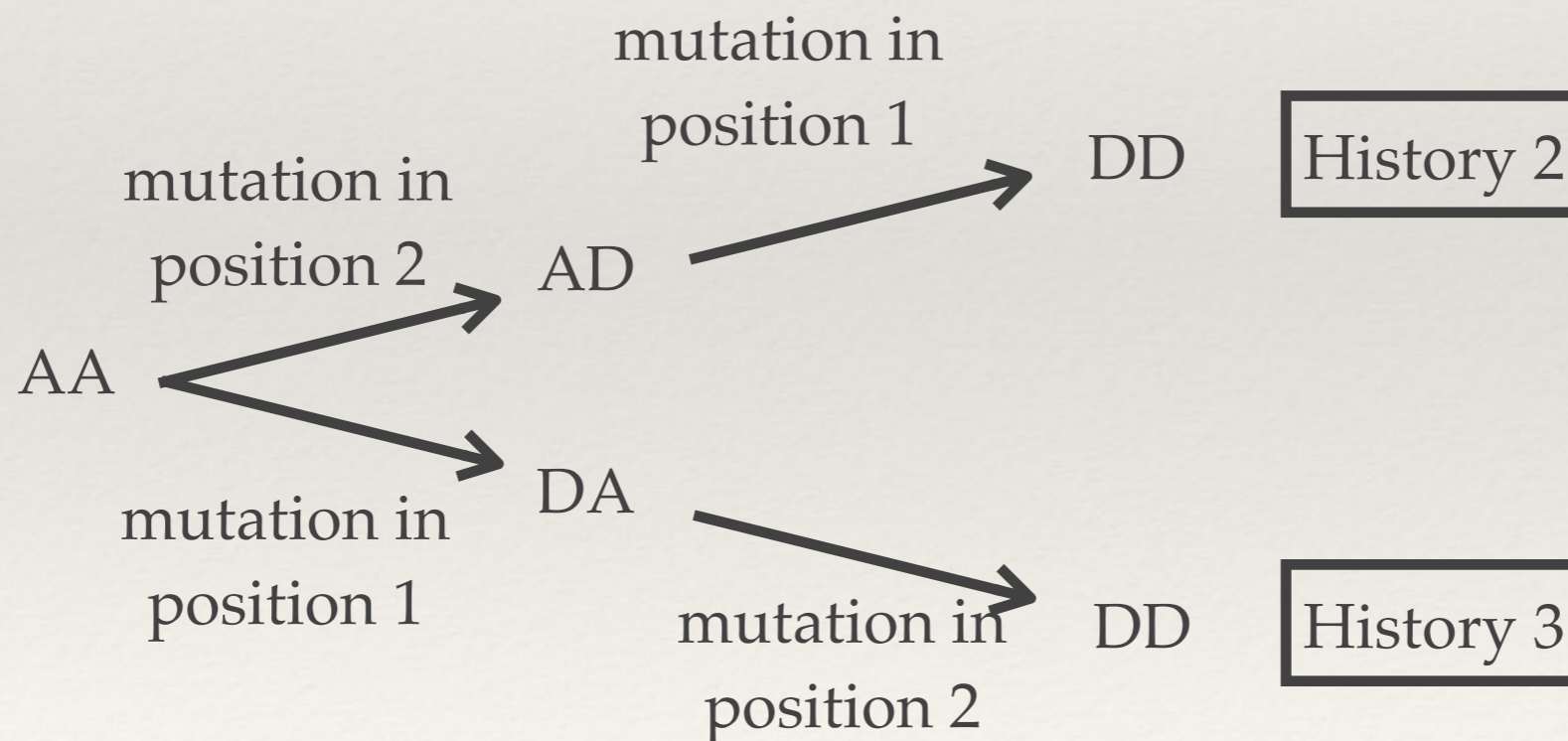
Methodology

- ❖ **The rule of the four haplotypes for two positions:**
Assuming no recurrent mutation and having no recombination (same genealogy), no more than three different haplotypes can be formed.



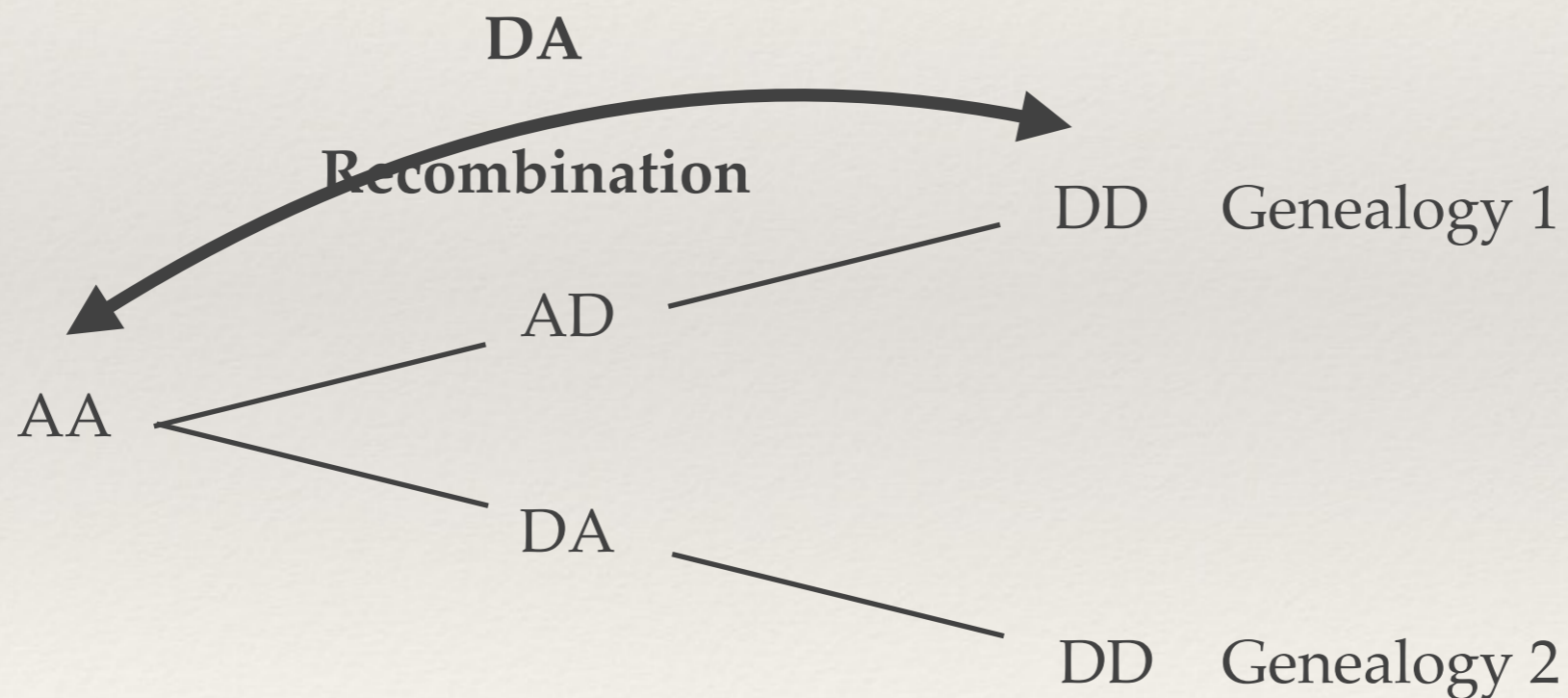
Methodology

- ❖ **The rule of the four haplotypes for two positions:**
Assuming no recurrent mutation and having no recombination (same genealogy), no more than three different haplotypes can be formed.



Methodology

- ❖ **The rule of the four haplotypes for two positions:**
Assuming no recurrent mutation and having no recombination (same genealogy), no more than three different haplotypes can be formed.



Methodology

- ❖ **The rule of the four haplotypes for two positions:**
Assuming no recurrent mutation and having no recombination (same genealogy), no more than three different haplotypes can be formed.
- ❖ As expected, all combinations producing incompatibilities between genealogies have the four possible haplotypes.
- ❖ No more than three populations (plus the outgroup) are necessary to observe the four haplotypes (that is, one haplotype per population).

Methodology

❖ Selecting the incompatible fragments:

```
+-----+
| GENERAL SUMMARY |
+-----+
Type      #positions

AAP       383
DDD      1129
PAP       33
DPP       10
DDP       91
APA      214
PAA      204
PPP       40
APP       8
DAP       3
PDP       8
PPA      24
AAD      18
DDA      14
DPD       1
PPD       2
DPA       6
APD       5
```

1. Look for all types of variants.
2. Find the incompatible combinations.
3. Sort each state by its position.
4. Assign the fragments that are incompatible with the contiguous.

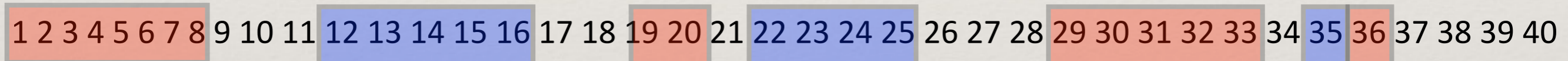
Methodology

- ❖ Selecting the incompatible fragments:

[9a] DPP 12,15,16,22,25,35,45,47,64

[9b] APD 3,4,5,8,19,20,29,32,33,36,54,58,72,90

3,4,5,8, 12,15,16,19,20, 22,25, 29,32,33, 35,36,45,47,54,58,64,72,90



Using all incompatibility combinations, we can have some overlapping:



Methodology: missing data

- ❖ Considering a weight factor for positions having missing data:

	PP	AD
pop1	A	A
	T	A
	T	A
	T	N
	N	N
pop2	A	T
	T	T
	N	T
	N	N
	N	N
out	A	A
	A	A
pos1		pos2

Methodology: missing data

- ❖ Consider a weight factor for positions having missing data:

	PP	AD	Incompatible Genealogies?
pop1	A	A	
	T	N	
	T	A	
	T	N	
	N	N	
pop2	A	T	
	T	T	
	N	T	
	N	N	
	N	N	
out	A	A	
	A	A	
pos1		pos2	

Methodology: missing data

- ❖ Consider a weight factor for positions having missing data:

	PP	AD
pop1	A	A
	T	A
	T	A
	T	N/T
	N	N/T
pop2	A	T
	T	T
	N	T
	N	N/A
	N	N/A
out	A	A
	A	A
pos1		pos2

Incompatible Genealogies?

Given the missing data, AD may be PD, or AP, or PP, and the order of the individuals within population is unknown, the genealogies would be compatible!

Methodology: missing data

- ❖ Similarity of missing data versus pooled data:

```

+-----+
| INCOMPATIBLE FRAGMENTS |
+-----+

[...end]      [start...]      comb1      comb2      w
1847      4428      DDPP      PADD      1.0
4428      5521      PADD      PDPP      1.0
5521      6786      PDPP      PADP      0.99
6786      8164      PADP      PPPP      0.99
8164      9163      PPPP      PDAA      1.0
10245     11552     PAPP      AAPD      0.96
11552     12316     AAPD      PAPP      0.99
13993     14080     PDPP      PAPD      0.9
14080     14527     PAPD      PDPP      0.99
14533     14861     PDPP      AAPD      0.9
14861     15104     AAPD      PAPP      0.99
15581     15701     PPPP      PDAP      1.0
16938     17290     PAPP      PADA      0.99
17290     17767     PADA      PDPD      0.99
18409     18681     PDPP      PADP      0.99
18681     22726     PADP      PDPP      1.0
22726     22961     PDPP      DAPA      0.56
22961     23511     DAPA      PFAP      1.0
24441     24799     AADP      AAPD      0.99
25946     27251     AAPD      PDPP      0.99
28820     29028     DDPP      AAPD      0.96
29292     29548     AAPD      DDPP      0.99
29548     29599     DDPP      AAPD      0.97
29742     30004     PDPD      PPDD      1.0
30004     30085     PPDD      DDPP      1.0
30552     30715     DDPP      AAPD      0.97

```

- ❖ The probability that we have in the entire sample a state (A or D) given the observation with missing data can be calculated:

$$P(\text{Ant} \mid \text{Ans}) + P(\text{Pnt} \mid \text{Ans}) = 1$$

$$P(\text{Dnt} \mid \text{Dns}) + P(\text{Pnt} \mid \text{Dns}) = 1$$

- ❖ Assuming a simple model of polymorphism versus divergence for each population, these probabilities are easily obtained using conditional probabilities and coalescent theory.

Results: Coalescent simulations

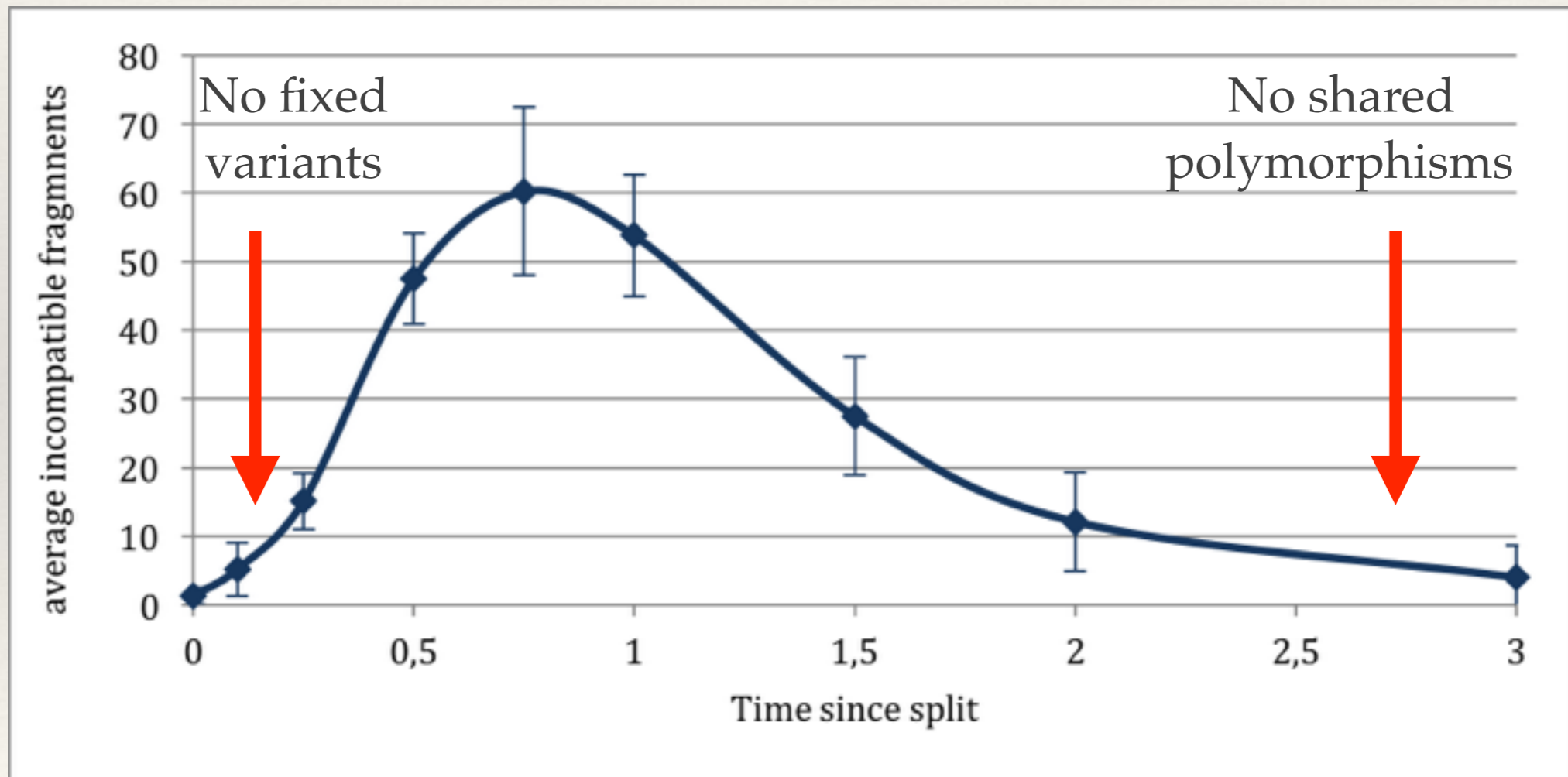
- ❖ Detection of recombinant events. True and false positive detection of Incompatible Genealogies:

VALIDATION

- ❖ No incompatible fragments were observed in simulations with no recombinations.
- ❖ In case using $R > 0$, we never find incompatible fragments in the same real tree (the real tree was obtained using the *check tree* function in *ms* software, which show all trees).

Results: Coalescent simulations

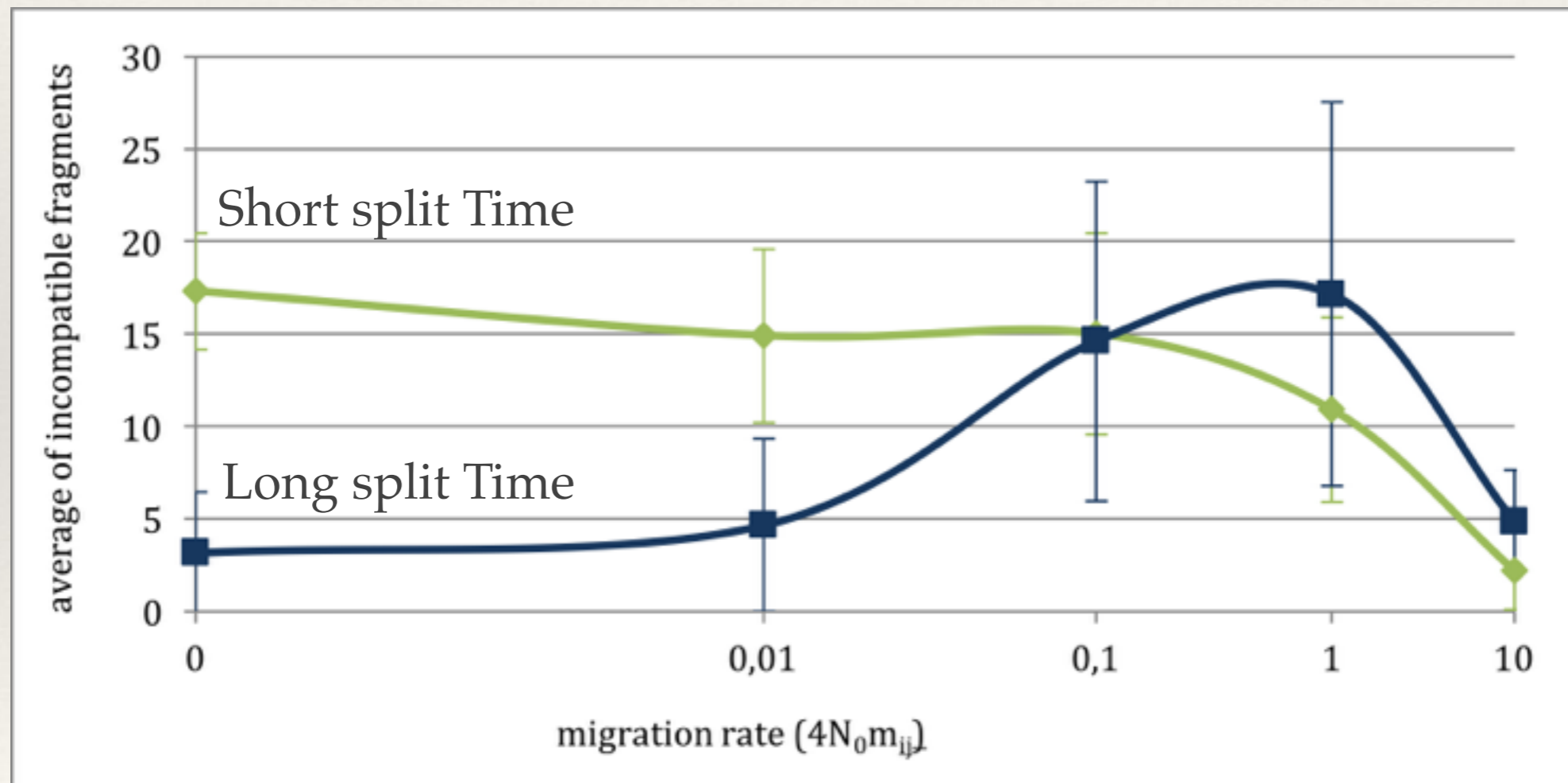
- ❖ The Time of Split among populations and the Detection of Incompatible Genealogies:



Relation between time since split (relative to $4N$ generations) between two populations and incompatible fragments found. No migration among populations.

Results: Coalescent simulations

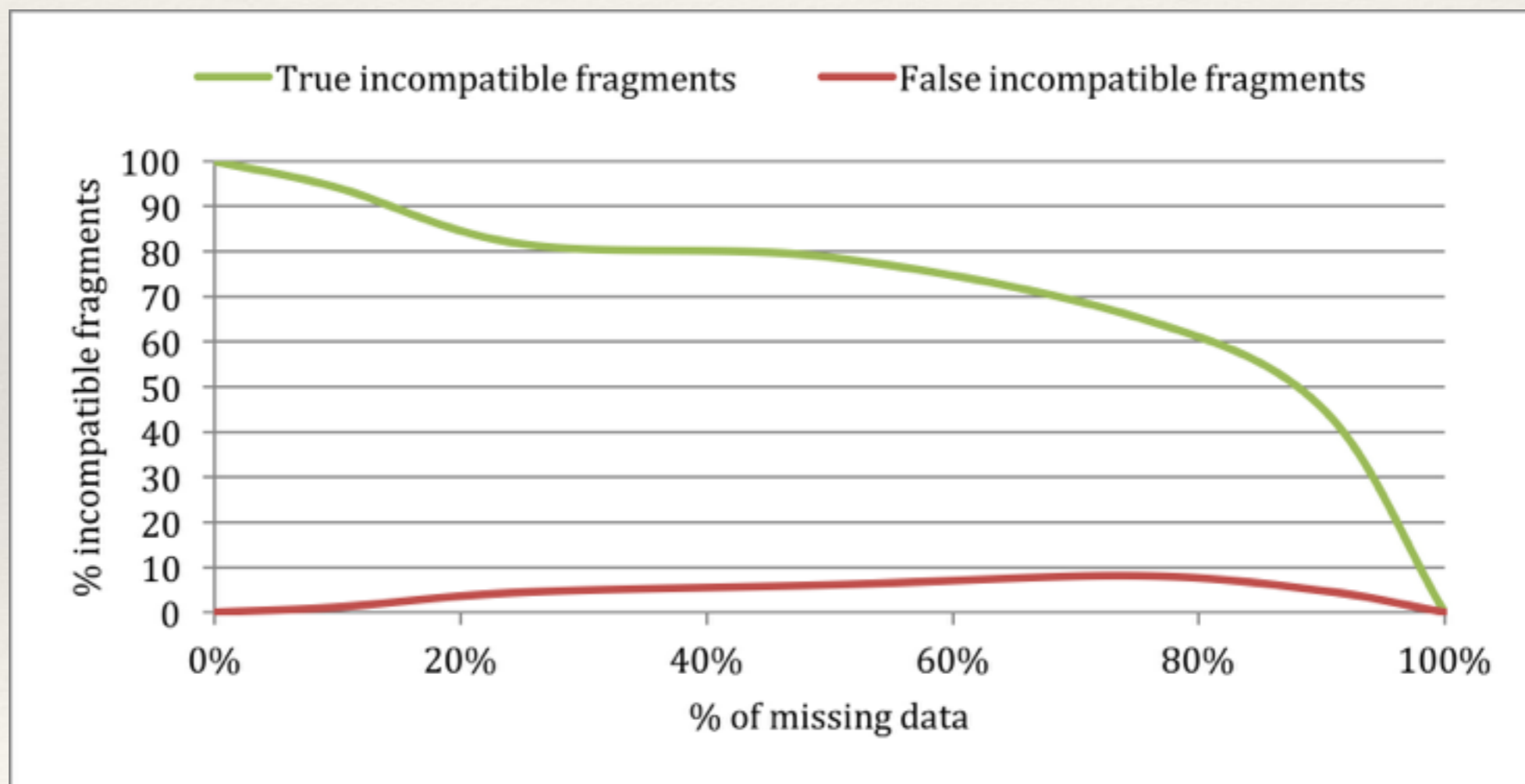
- ❖ The Migration parameter and the Detection of Incompatible Genealogies:



Relation between different migration rates ($4N_0m_{ij}$) and average number of incompatible fragments. Analysis done in two populations, with unidirectional migration. Green: short time ($0.25 \cdot 4N$ generations) since populations' split. Blue: long time ($3 \cdot 4N$ generations) since populations' split.

Results: Coalescent simulations

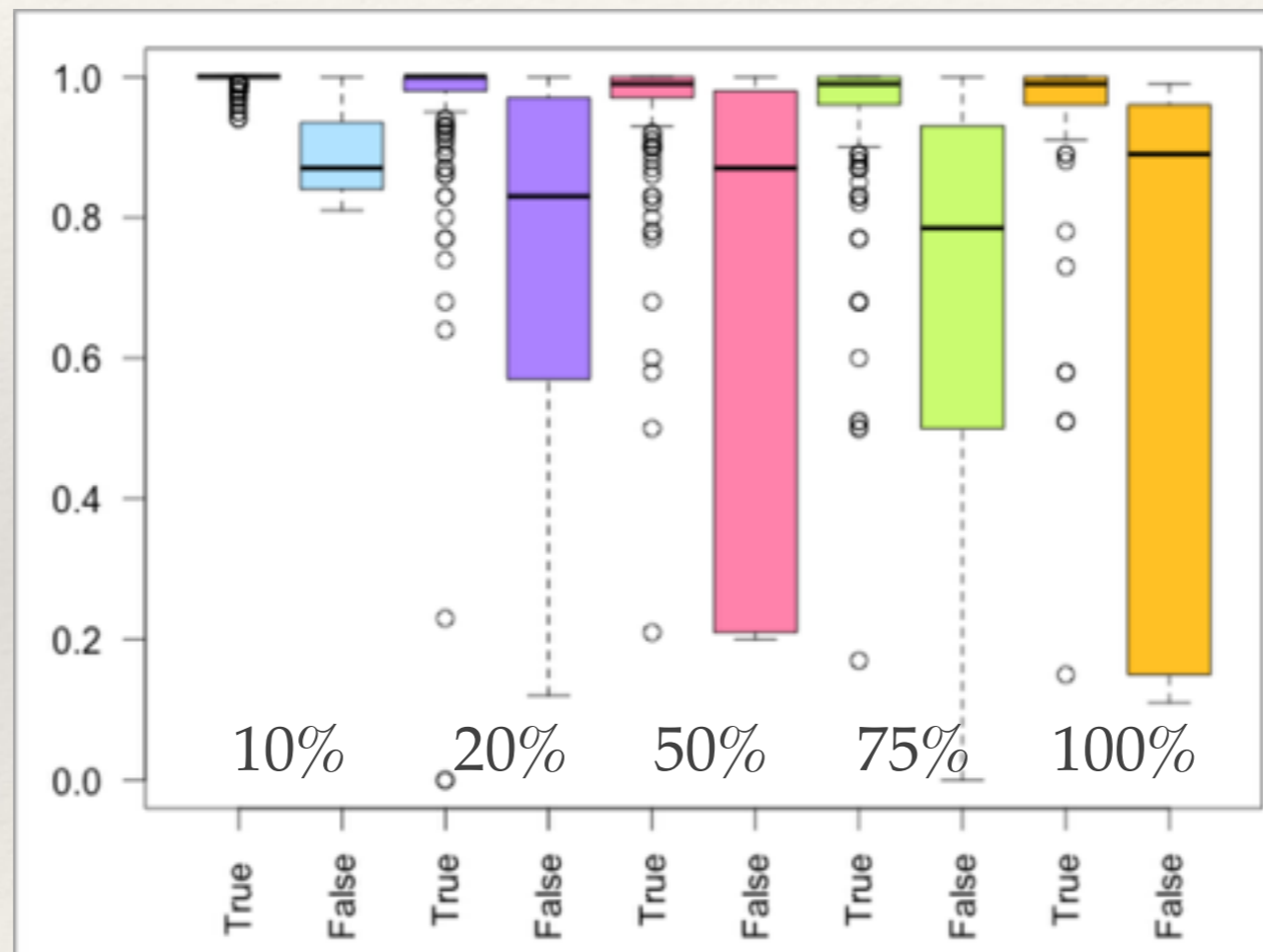
- ❖ The Missing data and the Detection of Incompatible Genealogies. True and False Positives:



Percentage of true and false incompatible fragments in different masks of missing data in relation to a sample with no missing data.

Results: Coalescent simulations

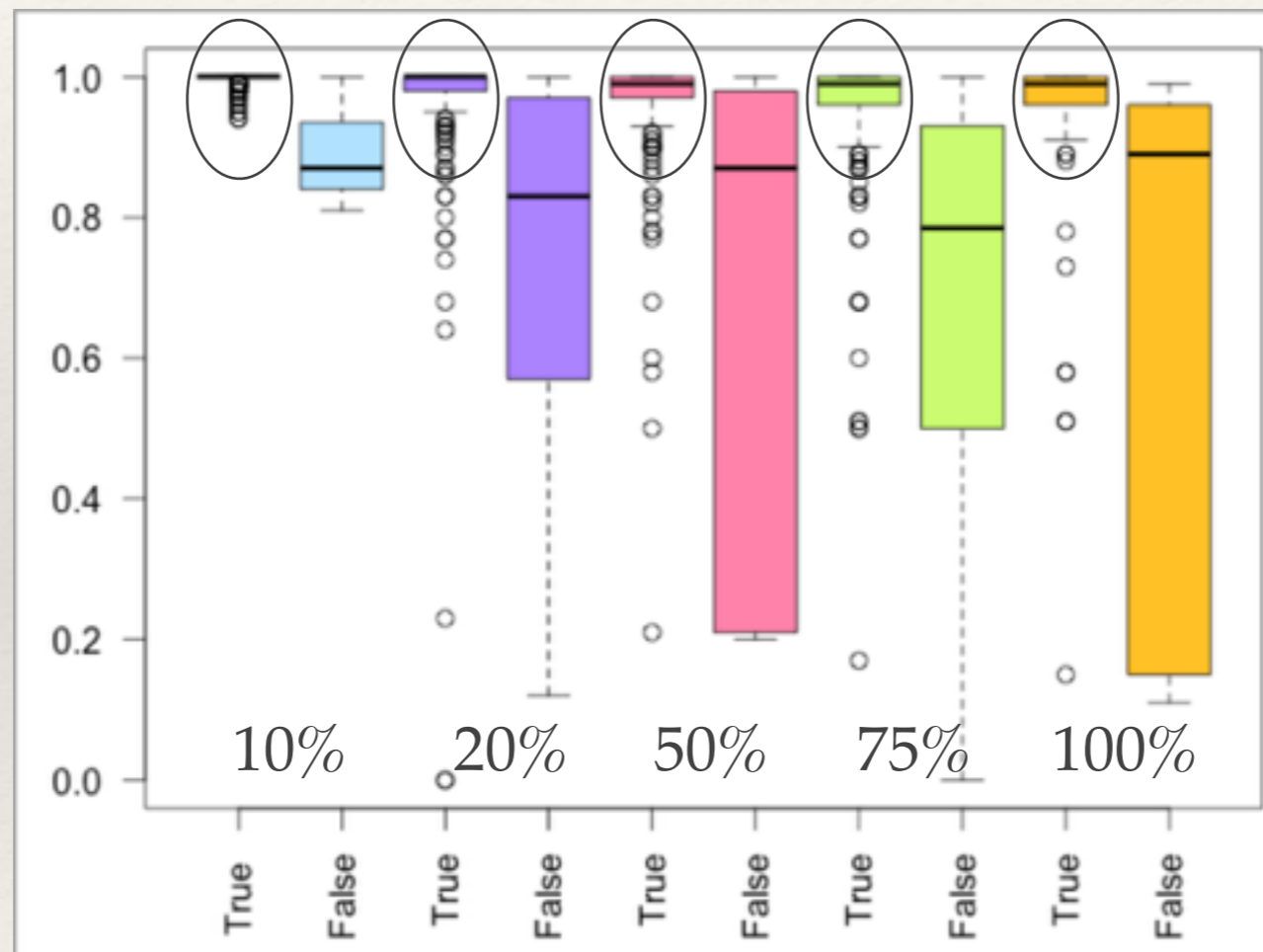
- ❖ The Missing data and the Detection of Incompatible Genealogies. The weight factor:



Boxplot of the normalised weight of reliability in true and false incompatible fragments for each mask with different percentage of missing data simulated. Percentage of missing in order: 10%, 25%, 50%, 75%, 90%.

Results: Coalescent simulations

- ❖ The Missing data and the Detection of Incompatible Genealogies. The weight factor:



Boxplot of the normalised weight of reliability in true and false incompatible fragments for each mask with different percentage of missing data simulated. Percentage of missing in order: 10%, 25%, 50%, 75%, 90%.

Results: Real data

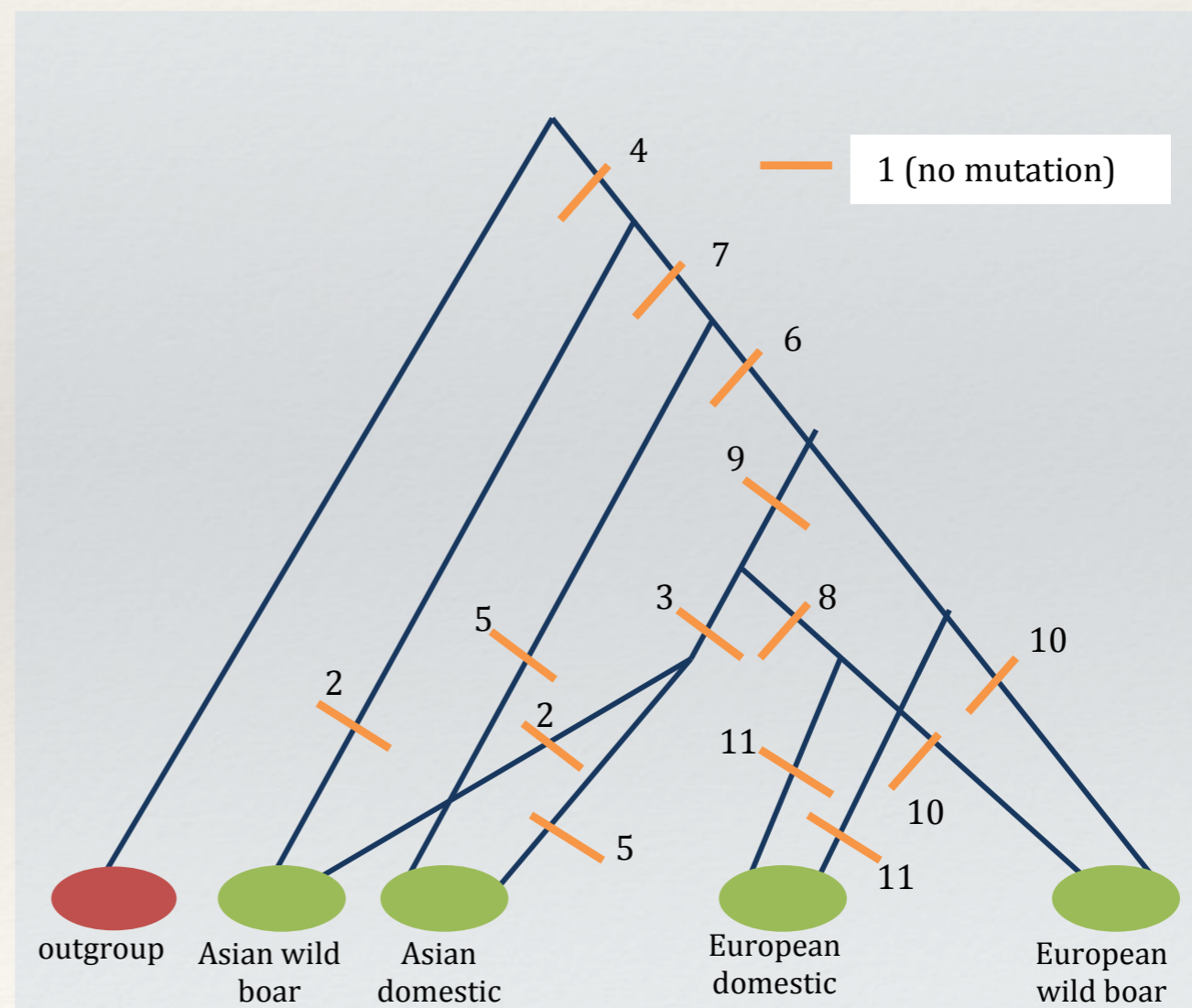
- ❖ Study of the variant sites along the chromosome 10 in four populations (around 10 samples each) of the species *Sus scrofa* (pig).
- ❖ The More General Tree and other frequent Tree Genealogies.
- ❖ The recombination rate and the length size of incompatible genealogies.
- ❖ The Distribution of Tree length genealogies.



Results: Real data

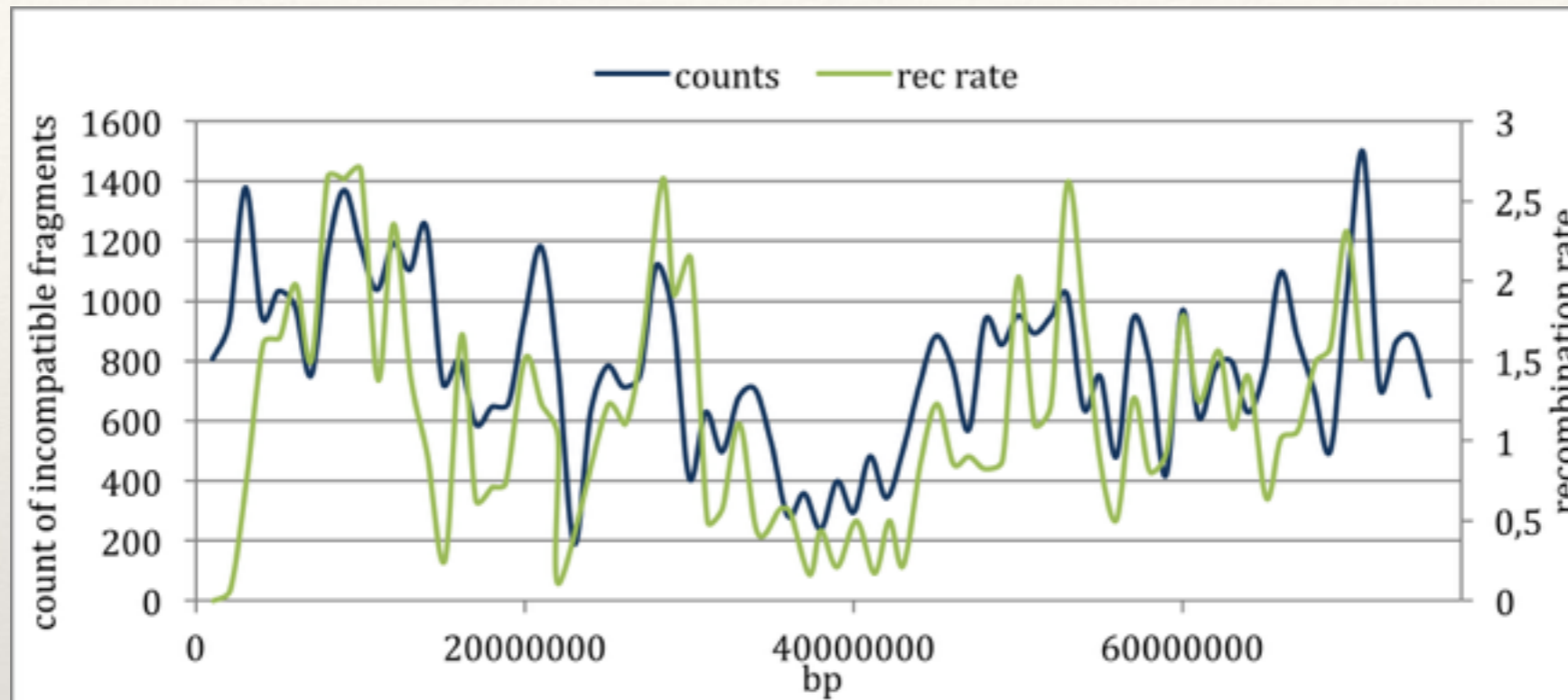
- ❖ The 11 more frequent type of combinations (85% variants) and their genealogical reconstruction.

#	Combination type	Counts
1	AAAA	51403963
2	PAAA	414110
3	PPAA	163531
4	DDDD	127040
5	APAA	71559
6	PPDD	64476
7	PDDD	38528
8	AAPP	31307
9	PPPP	29767
10	AAAP	28245
11	AAPA	24377



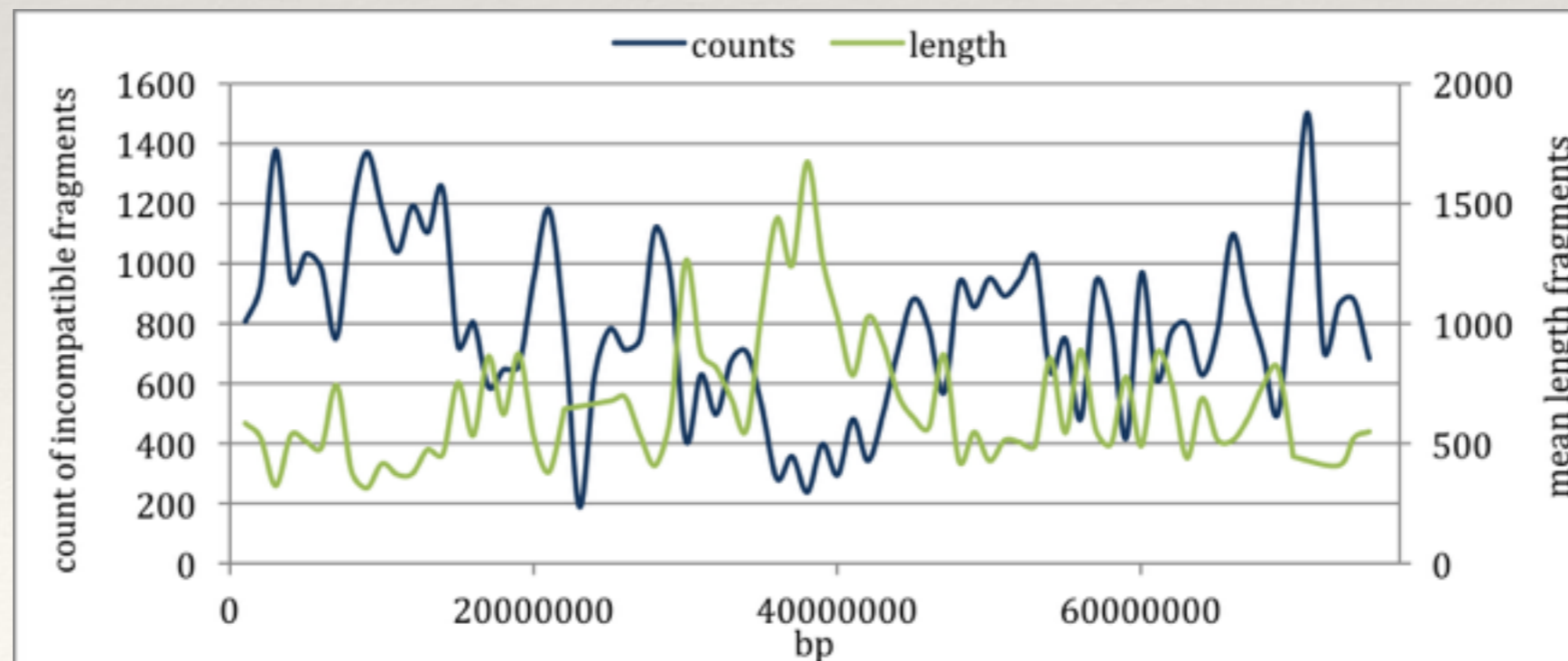
Results: Real data

Number of
Incompatible
fragments



Recombination
rate

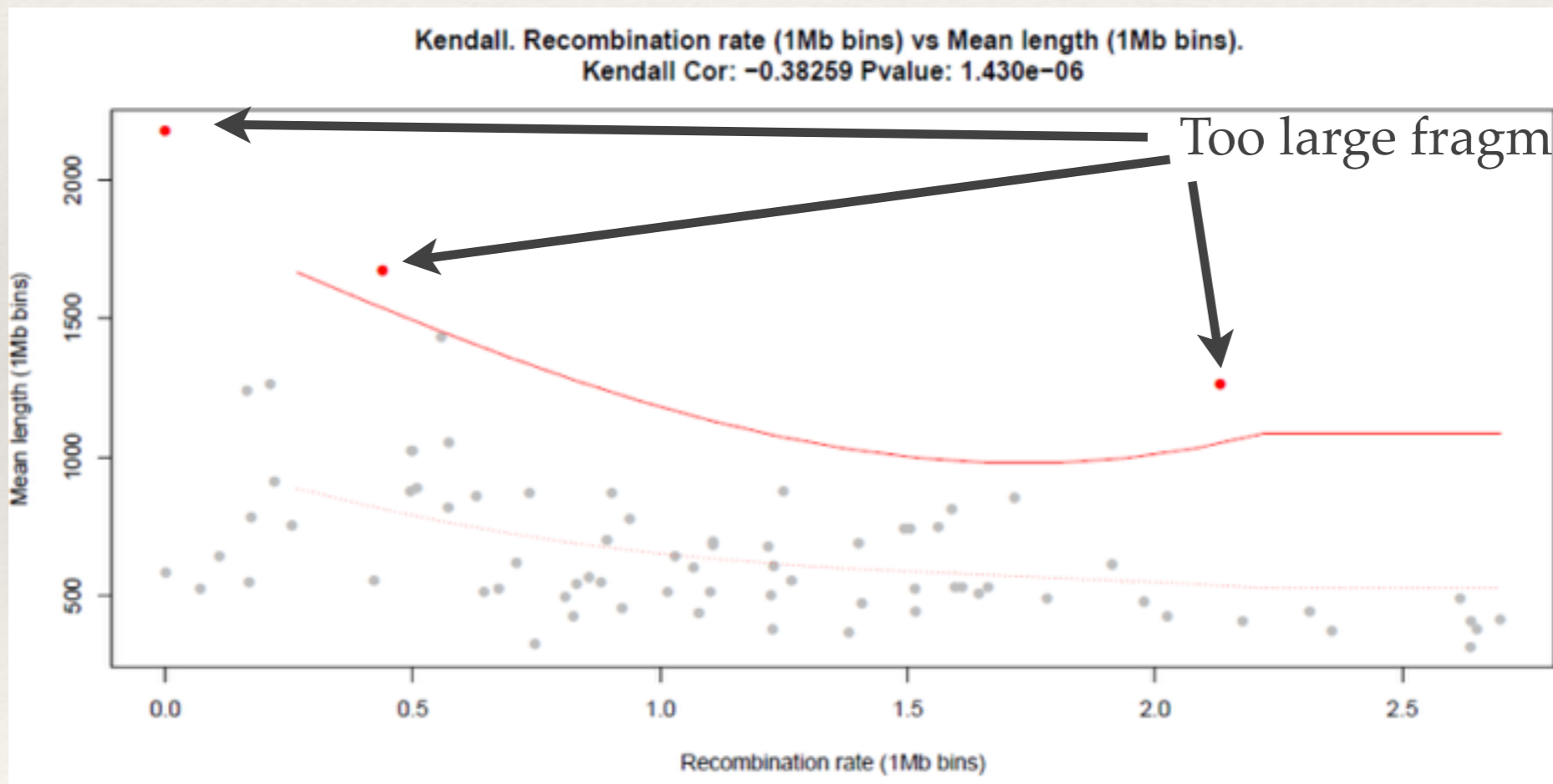
Number of
Incompatible
fragments



Mean length of
Incompatible
fragments

Results: Real data

- ❖ Comparison between lengths of incompatible fragments and recombination rate. Empirically, we find few outliers.



Mean length of
Incompatible
fragments

Recombination
rate

Perspectives

- ❖ Useful for discretising the genome into non-incompatible windows when using a sliding windows analysis.
- ❖ Useful for counting all different branches appearing in the sample and reconstructing the history of the species for the whole and at each genomic region.
- ❖ Factors used for weighting missing data: consider other weights. For example use the number of incompatible variants versus contiguous fragments as a factor for the reliability of the incompatibility.

Perspectives

- ❖ Detection of local evolutionary events:
 - ❖ Relationship between recombination rate and number and length of incompatible genealogies. The NO fit of recombination map versus patterns of incompatible genealogies observation can be caused by additional evolutionary processes.
 - ❖ An excess of a given type of a variant (a mutation in a specific branch) in some regions may be unexpected under the general genealogical pattern, which may indicate a rare evolutionary process. Study the distribution of variant types and the distribution of incompatible fragment lengths versus different evolutionary models.
 - ❖ Combination with other methodologies (for example D-statistic).
 - ❖ A HMM may be constructed for differentiating regions having migration from each popA to each popB, or no migration, considering the incompatible genealogical regions.

Software: DIGUP

<https://github.com/mvidalv/DIGUP>

The screenshot shows a web browser window displaying the GitHub repository page for 'mvidalv / DIGUP'. The browser's address bar shows the URL 'https://github.com/mvidalv/DIGUP'. The repository page includes a navigation bar with 'This repository', 'Search', 'Pull requests', 'Issues', 'Marketplace', and 'Explore'. The repository name 'mvidalv / DIGUP' is displayed, along with 'Watch' (1), 'Star' (0), and 'Fork' (0) buttons. Below the repository name, there are tabs for 'Code', 'Issues' (0), 'Pull requests' (0), 'Projects' (0), 'Wiki', and 'Insights'. The repository title is 'Detection of Incompatible Genealogies for Unphased Populations'. A summary bar shows '5 commits', '1 branch', '0 releases', and '0 contributors'. Below this, there are buttons for 'Branch: master', 'New pull request', 'Create new file', 'Upload files', 'Find file', and 'Clone or download'. The commit history shows a single commit by Mireia Vidal: 'manual added' on Aug 31, 2015. The file list includes 'DIGUP.py', 'digup_manual.pdf', and 'readme.md'. The 'readme.md' file is selected, showing its content: 'DIGUP is a program that detects incompatible genealogies among populations for unphased data.'

5 commits 1 branch 0 releases 0 contributors

Branch: master New pull request Create new file Upload files Find file Clone or download

Mireia Vidal manual added Latest commit 2f2232e on Aug 31, 2015

DIGUP.py	DIGUP	2 years ago
digup_manual.pdf	manual added	2 years ago
readme.md	DIGUP	2 years ago

readme.md

DIGUP is a program that detects incompatible genealogies among populations for unphased data.

Software: DIGUP

<https://github.com/mvidalv/DIGUP>

DIGUP usage

DIGUP is able to read both *fasta* and *ms* format and includes several arguments, ones are optional and others are required. Below, DIGUP usage and detailed explanation of each argument.

```
Usage: DIGUP.py input_file -n n -i i1 i2.. ip [-o {1,2,12}] [-ms]
[-l length] [-nt nt1 nt2.. ntp] [-G]
```

- n** total number of sequences (including the outgroup).
- i** total number of individuals in each population (in the same order as in the input file). Last population is considered to be the outgroup.
- ms** if input file is in *ms* format (default reading is for *fasta* format)
- o** output type (1, 2 or both as 12) for the classification of variant positions. Default output type is 1, which includes classification, of each population, of all variant positions. Output type 2 includes same

DIGUP project

Mireia Vidal-Villarejo (Hohenheim U., Germany)
Luca Ferretti (Pirbright I., UK)
Sebastian E. Ramos-Onsins (PI)



ngasp core team

Jordi Leno-Colorado (co-directed PhD in Genetics)
Joan Jené (Computer Scientist Engineer)
Gonzalo Vera (Head Engineer)
Sebastian E. Ramos-Onsins (PI)

Luca Ferretti (Pirbright I., UK)
Javier Navarro (Comp. Sc., PCB)
Carlos Montemuiño (co-directed PhD in Comp. Sc.)
Sara Guirao-Rico (CRAG)
Miguel Pérez-Enciso (ICREA-CRAG)

Julio Rozas (UB-Barcelona)
Alejandro Sánchez-Gracia (UB-Barcelona)
Porfidio Hernández-Budé (UAB-Barcelona)
Emanuele Raineri (CNAG-Barcelona)

