SVDQuartets Tutorial

Laura Kubatko and Dave Swofford

Slide credit: Dave Swofford

June 4, 2018

What SDVQuartets is ....

- **SVDQuartets** is a method for species tree inference based on the multispecies coalescent that can be applied to multilocus, SNP, or coalescent independent sites data.
- The theory underlying the method is valid for data arising from very general models:
  - GTR+I+G model and all submodels
  - with or without the molecular clock
  - variation in rates and effective population sizes along branches
  - gene flow between sister taxa

What SDVQuartets is not ....

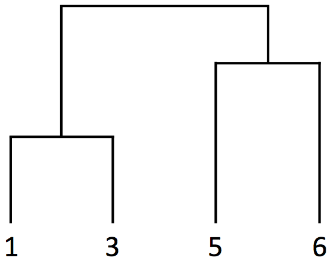
- **SVDQuartets is not a concatenation method!**
- It is NOT a summary statistics method – at least not in the traditional way.
- It does NOT try to approximate maximum likelihood (or anything else).
- It is NOT model-free.

Why is it called SVDQuartets?

- **Basic idea:**
  - Break the problem into quartets = sets of 4 taxa
  - Infer the unrooted four-taxon tree for each quartet – this is done using a mathematical technique called singular value decomposition (SVD)
  - Reassemble the quartets to form an overall species tree estimate
- Lots of small details for each step – we'll review the major ideas in the first half of today's tutorial

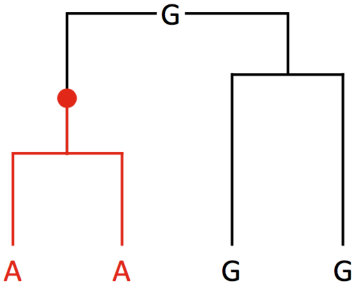
Site pattern frequencies

- **Example:**

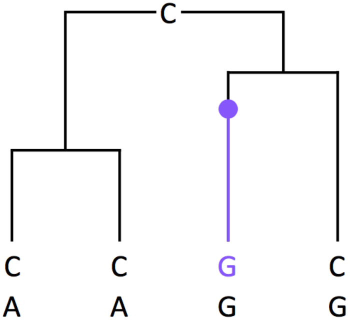


A tree for 4 taxa, which may be a subtree of a larger tree

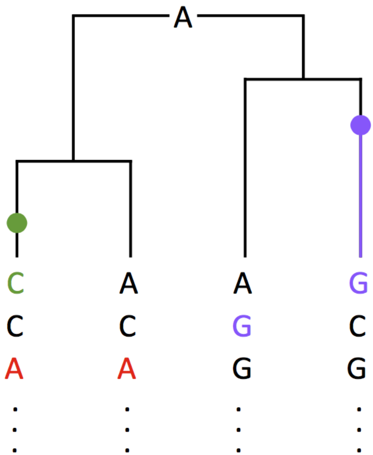
Site pattern frequencies



Site pattern frequencies

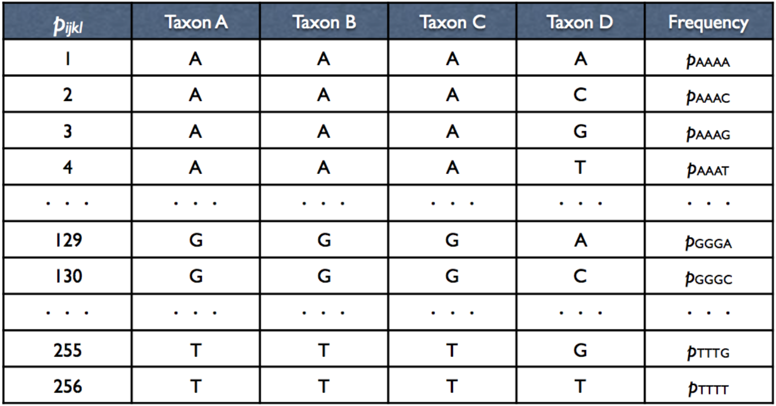


Site pattern frequencies



Site pattern frequencies

- **For each set of 4 sequences (quartet), we can count the relative frequencies of the 256 possible site patterns**



p_{ijkl}	Taxon A	Taxon B	Taxon C	Taxon D	Frequency
1	A	A	A	A	p_{AAAA}
2	A	A	A	C	p_{AAAC}
3	A	A	A	G	p_{AAAG}
4	A	A	A	T	p_{AAAT}
. . .	. . .	. . .	. . .	. . .	. . .
129	G	G	G	A	p_{GGGA}
130	G	G	G	C	p_{GGGC}
. . .	. . .	. . .	. . .	. . .	. . .
255	T	T	T	G	p_{TTTG}
256	T	T	T	T	p_{TTTT}

Flattening matrix

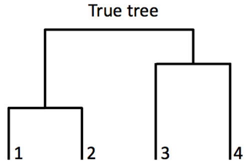
- **For each set of 4 sequences (quartet), we represent the pattern frequencies by a flattening matrix:**

$$\mathrm{Flat}_{\{1,3\},\{2,4\}}(P) = \begin{array}{cc} & \begin{array}{ccccccc} \texttt{AA} & \texttt{AC} & \texttt{AG} & \texttt{AT} & \texttt{CA} & \texttt{CC} & \cdots \end{array} \\ \begin{array}{c} \texttt{AA} \\ \texttt{AC} \\ \texttt{AG} \\ \texttt{AT} \\ \texttt{CA} \\ \vdots \end{array} & \begin{pmatrix} p_{\texttt{AAAA}} & p_{\texttt{AAAC}} & p_{\texttt{AAAG}} & p_{\texttt{AAAT}} & p_{\texttt{ACAA}} & p_{\texttt{ACAC}} & \cdots \\ p_{\texttt{AACA}} & p_{\texttt{AACC}} & p_{\texttt{AACG}} & p_{\texttt{AACT}} & p_{\texttt{ACCA}} & p_{\texttt{ACCC}} & \cdots \\ p_{\texttt{AAGA}} & p_{\texttt{AAGC}} & p_{\texttt{AAGG}} & p_{\texttt{AAGT}} & p_{\texttt{ACGA}} & p_{\texttt{ACGC}} & \cdots \\ p_{\texttt{AATA}} & p_{\texttt{AATC}} & p_{\texttt{AATG}} & p_{\texttt{AATT}} & p_{\texttt{ACTA}} & p_{\texttt{ACTC}} & \cdots \\ p_{\texttt{CAAA}} & p_{\texttt{CAAC}} & p_{\texttt{CAAG}} & p_{\texttt{CAAT}} & p_{\texttt{CCAA}} & p_{\texttt{CCAC}} & \cdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \end{pmatrix} \end{array}.$$

- **Matrix rank:** number of linearly independent rows and columns

- **Main result:** when the flattening corresponds to the tree that generated the data, the matrix rank will be *fewer* than the number of rows/columns

Intuition on reduced rank/linear dependencies



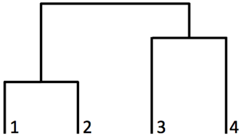
A	C	A	C
A	C	C	A
C	A	A	C
C	A	C	A

E.g., all 4 of these site patterns have the same expected frequency

f(AC|AC)=f(AC|CA)=f(CA|AC)=f(CA|CA)

Intuition on reduced rank/linear dependencies

True tree

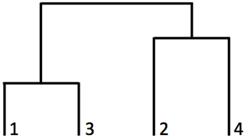


A	C	A	C
A	C	C	A
C	A	A	C
C	A	C	A

E.g., all 4 of these site patterns have the same expected frequency

f(AC|AC)=f(AC|CA)=f(CA|AC)=f(CA|CA)

Incorrect tree

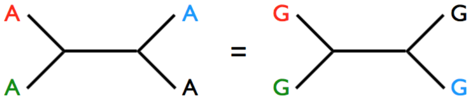


A	A	C	C
A	C	C	A
C	A	A	C
C	C	A	A

These patterns are **not** all expected to have the same expected frequency *if they evolved on the other tree*

f(AA|CC)≠f(AC|CA)≠f(CA|AC)≠f(CC|AA)

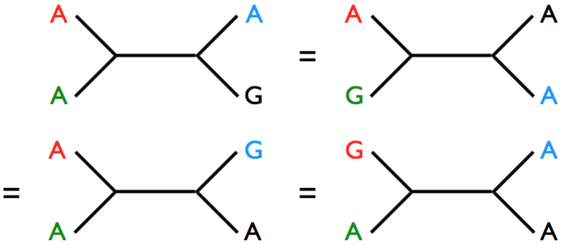
Intuition on reduced rank/linear dependencies



Flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	*a*			
AG				
GA				
GG				*a*

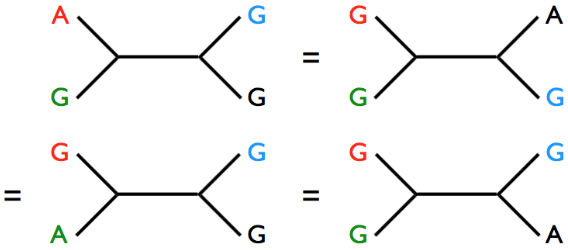
Intuition on reduced rank/linear dependencies



Flattening matrix
for 1,2|3,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	
AG	*b*			
GA	*b*			
GG				*a*

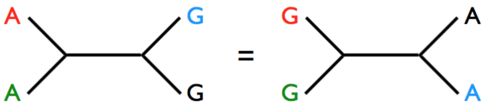
Intuition on reduced rank/linear dependencies



Flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	
AG	*b*			*b*
GA	*b*			*b*
GG		*b*	*b*	*a*

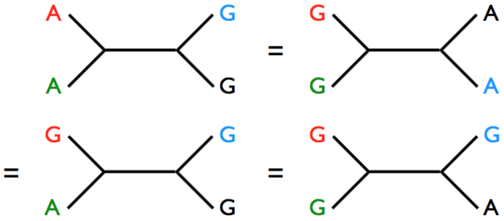
Intuition on reduced rank/linear dependencies



Flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	c
AG	*b*			*b*
GA	*b*			*b*
GG	c	*b*	*b*	*a*

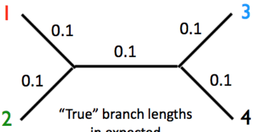
Intuition on reduced rank/linear dependencies



Flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	*c*
AG	*b*	*d*	*d*	*b*
GA	*b*	*d*	*d*	*b*
GG	*c*	*b*	*b*	*a*

Numerical example



“True” branch lengths
in expected
substitutions/site

Expected flattening matrix for 1,2|3,4

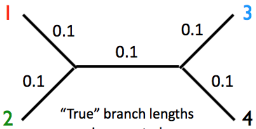
	AA	AG	GA	GG
AA	0.093008	0.061355	0.061355	0.068115
AG	0.061355	0.046728	0.046728	0.061355
GA	0.061355	0.046728	0.046728	0.061355
GG	0.068115	0.061355	0.061355	0.093008

Expected site-pattern
frequencies

p_{AAAA}	0.09300841
p_{AAAG}	0.06135527
p_{AAGA}	0.06135527
p_{AAGG}	0.06811487
p_{AGAA}	0.06135527
p_{AGAG}	0.04672782
p_{AGGA}	0.04672782
p_{AGGG}	0.06135527
p_{GAAA}	0.06135527
p_{GAAG}	0.04672782
p_{GAGA}	0.04672782
p_{GAGG}	0.06135527
p_{GGAA}	0.06811487
p_{GGAG}	0.06135527
p_{GGGA}	0.06135527
p_{GGGG}	0.09300841

etc.

Numerical example



"True" branch lengths in expected substitutions/site

Expected site-pattern frequencies

p_{AAAA}	0.09300841
p_{AAAG}	0.06135527
p_{AAGA}	0.06135527
p_{AAGG}	0.06811487
p_{AGAA}	0.06135527
p_{AGAG}	0.04672782
p_{AGGA}	0.04672782
p_{AGGG}	0.06135527
p_{GAAA}	0.06135527
p_{GAAG}	0.04672782
p_{GAGA}	0.04672782
p_{GAGG}	0.06135527
p_{GGAA}	0.06811487
p_{GGAG}	0.06135527
p_{GGGA}	0.06135527
p_{GGGG}	0.09300841

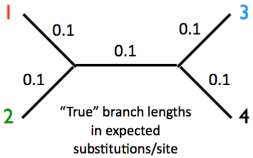
Expected flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	0.093008	0.061355	0.061355	0.068115
AG	0.061355	0.046728	0.046728	0.061355
GA	0.061355	0.046728	0.046728	0.061355
GG	0.068115	0.061355	0.061355	0.093008

Delete redundant 3rd row and column...

	AA	AG	GG
AA	0.093008	0.061355	0.068115
AG	0.061355	0.046728	0.061355
GG	0.068115	0.061355	0.093008

Numerical example



Expected site-pattern frequencies

p_{AAAA}	0.09300841
p_{AAAG}	0.06135527
p_{AAGA}	0.06135527
p_{AAGG}	0.06811487
p_{AGAA}	0.06135527
p_{AGAG}	0.04672782
p_{AGGA}	0.04672782
p_{AGGG}	0.06135527
p_{GAAA}	0.06135527
p_{GAAG}	0.04672782
p_{GAGA}	0.04672782
p_{GAGG}	0.06135527
p_{GGAA}	0.06811487
p_{GGAG}	0.06135527
p_{GGGA}	0.06135527
p_{GGGG}	0.09300841

Expected flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	0.093008	0.061355	0.061355	0.068115
AG	0.061355	0.046728	0.046728	0.061355
GA	0.061355	0.046728	0.046728	0.061355
GG	0.068115	0.061355	0.061355	0.093008

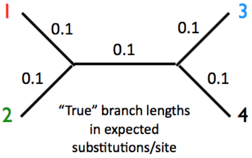
Delete redundant 3rd row and column...

	AA	AG	GG
AA	0.093008	0.061355	0.068115
AG	0.061355	0.046728	0.061355
GG	0.068115	0.061355	0.093008

Note that we can now obtain the last column of the above matrix as a linear combination of the first two columns:

$$f_{AA,GG} = -f_{AA,AA} + 2.62617\, f_{AA,AG} = 0.068115$$
$$f_{AG,GG} = -f_{AG,AA} + 2.62617\, f_{AG,AG} = 0.061355$$
$$f_{GG,GG} = -f_{GG,AA} + 2.62617\, f_{GG,AG} = 0.093008$$

Numerical example



Expected site-pattern frequencies

p_{AAAA}	0.09300841
p_{AAAG}	0.06135527
p_{AAGA}	0.06135527
p_{AAGG}	0.06811487
p_{AGAA}	0.06135527
p_{AGAG}	0.04672782
p_{AGGA}	0.04672782
p_{AGGG}	0.06135527
p_{GAAA}	0.06135527
p_{GAAG}	0.04672782
p_{GAGA}	0.04672782
p_{GAGG}	0.06135527
p_{GGAA}	0.06811487
p_{GGAG}	0.06135527
p_{GGGA}	0.06135527
p_{GGGG}	0.09300841

Expected flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	0.093008	0.061355	0.061355	0.068115
AG	0.061355	0.046728	0.046728	0.061355
GA	0.061355	0.046728	0.046728	0.061355
GG	0.068115	0.061355	0.061355	0.093008

Delete redundant 3rd row and column...

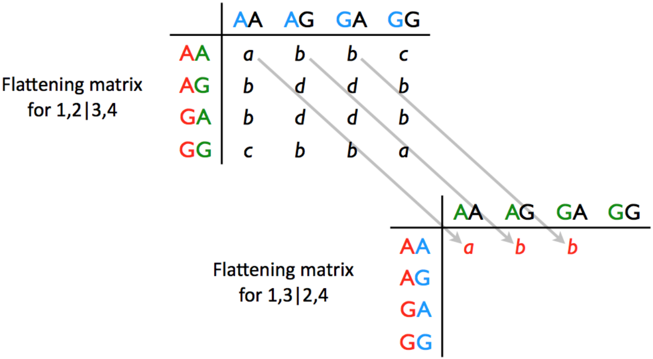
	AA	AG	GG
AA	0.093008	0.061355	0.068115
AG	0.061355	0.046728	0.061355
GG	0.068115	0.061355	0.093008

Note that we can now obtain the last column of the above matrix as a linear combination of the first two columns:

$$f_{AA,GG} = -f_{AA,AA} + 2.62617\, f_{AA,AG} = 0.068115$$
$$f_{AG,GG} = -f_{AG,AA} + 2.62617\, f_{AG,AG} = 0.061355$$
$$f_{GG,GG} = -f_{GG,AA} + 2.62617\, f_{GG,AG} = 0.093008$$

∴ ***matrix*** **has** ***only two linearly independent rows and columns; rank is 2***

What if we construct a flattening matrix for a tree that did NOT generate the data?



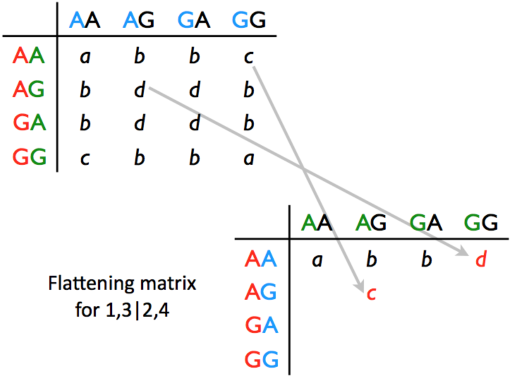
Flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	*c*
AG	*b*	*d*	*d*	*b*
GA	*b*	*d*	*d*	*b*
GG	*c*	*b*	*b*	*a*

Flattening matrix for 1,3|2,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	
AG				
GA				
GG				

What if we construct a flattening matrix for a tree that did NOT generate the data?



Flattening matrix for 1,2|3,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	*c*
AG	*b*	*d*	*d*	*b*
GA	*b*	*d*	*d*	*b*
GG	*c*	*b*	*b*	*a*

Flattening matrix for 1,3|2,4

	AA	AG	GA	GG
AA	*a*	*b*	*b*	*d*
AG		*c*		
GA				
GG				

What if we construct a flattening matrix for a tree that did NOT generate the data?

Flattening matrix for 1,2|3,4

$$\begin{array}{c|cccc} & \mathrm{AA} & \mathrm{AG} & \mathrm{GA} & \mathrm{GG} \\ \hline \mathrm{AA} & a & b & b & c \\ \mathrm{AG} & b & d & d & b \\ \mathrm{GA} & b & d & d & b \\ \mathrm{GG} & c & b & b & a \end{array}$$

Flattening matrix for 1,3|2,4

$$\begin{array}{c|cccc} & \mathrm{AA} & \mathrm{AG} & \mathrm{GA} & \mathrm{GG} \\ \hline \mathrm{AA} & a & b & b & d \\ \mathrm{AG} & b & c & d & b \\ \mathrm{GA} & b & d & c & b \\ \mathrm{GG} & d & b & b & a \end{array}$$

No redundant rows; matrix is full rank (=4)

How can we use this for species tree inference?

- **Fact:** Under the multispecies coalescent model for DNA sequence data:
  - the flattening matrix corresponding to the true tree has rank 10
  - the flattening matrix corresponding to each of the two alternative topologies has rank 16 (there are 16 rows and 16 columns)

- **Complication:** For empirical data, the site pattern frequencies approximate the true probabilities, but aren't exact

- **Solution:** Find a a way to measure how close we are to a reduced rank matrix
  Use singular value decomposition!

Singular value decomposition

- **Basic idea:** Decompose an initial matrix into 3 new ones, such that multi plying the new matrices as shown below returns the original matrix exactly

$$\mathbf{M} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}'$$

- The matrix $\boldsymbol{\Sigma}$ contains the **singular values** (16 values here, since the flattening matrix is 16×16)

- **Fact:** The number of non-zero singular values is equivalent to the matrix rank

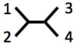
The SVD Score

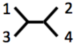
$$score = \sqrt{\sum_{i=5}^{16} s_i^2}$$

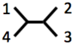
= "Frobenius distance" to nearest rank 4 matrix

Simulation conditions:

- tree = (((1:0.05,2:0.05):0.05,3:0.1):0.05,4:0.15)
- 1,000,000 sites
- HKY model: κ=4 π=(0.1, 0.2, 0.3, 0.4)
- all sites share same history (no incomplete lineage sorting, horizontal transfer, gene duplication and loss, etc.)







SV (*s*)	1,2 I 3,4	1,3 I 2,4	1,4 I 2,3
1	0.279686	0.278714	0.278716
2	0.21899	0.219191	0.219191
3	0.10902	0.110392	0.110389
4	0.056873	0.05709	0.05709
5	8E-05	0.006875	0.006886
6	6.14E-05	0.006315	0.006305
7	4.93E-05	0.003286	0.003286
8	3.8E-05	0.003244	0.003246
9	3.26E-05	0.002905	0.002903
10	3.09E-05	0.002499	0.002499
11	2.69E-05	0.001471	0.001472
12	2.23E-05	0.001182	0.001181
13	1.3E-05	0.001009	0.001008
14	1.03E-05	0.000937	0.000937
15	6.19E-06	0.000382	0.000384
16	1.56E-06	0.000377	0.000375
score	0.000133	0.011353	0.011354

More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

$$\left.\begin{array}{c} 12|34 \\ 12|35 \\ 12|45 \\ 14|35 \\ 23|45 \end{array}\right\}$$

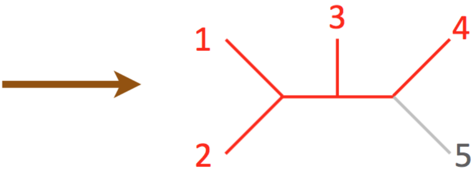
Suppose we infer these quartet relationships for 5 taxa

More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45

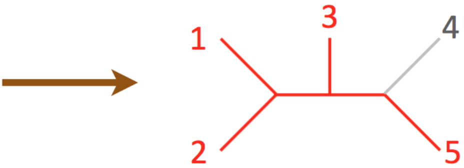


More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45

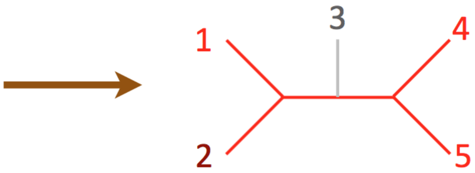


More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45

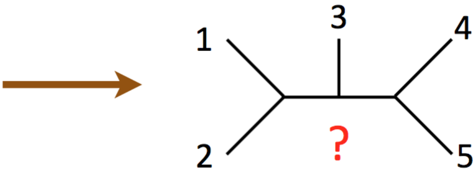


More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45

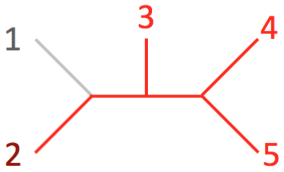


More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45

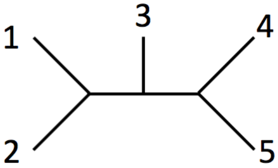


More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45



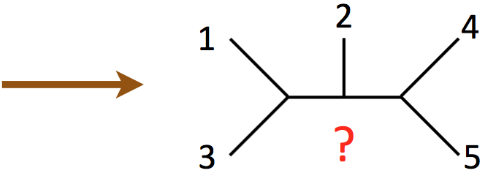
4 consistent quartets, 1 inconsistent quartet

More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45

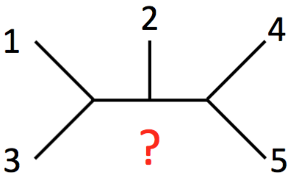


More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45

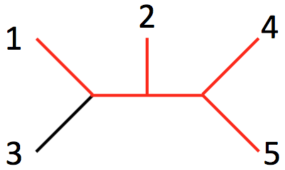


More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

12|34
12|35
12|45
14|35
23|45



More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

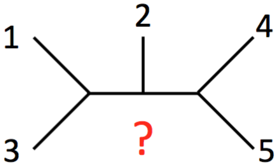
12|34

12|35

12|45

14|35

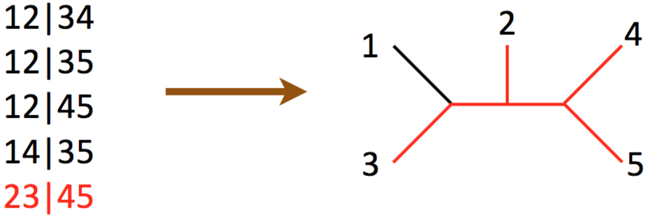
23|45



More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).



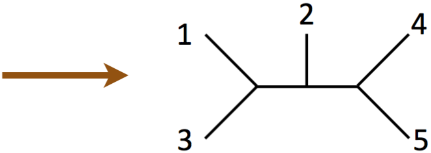
12|34
12|35
12|45
14|35
23|45

More than 4 taxa

Compute invariant scores for all quartets, choosing the best resolution for each one.

Search for a tree that minimizes the number of *inconsistent quartets* (i.e., seek a solution to the Maximum Quartet Consistency problem).

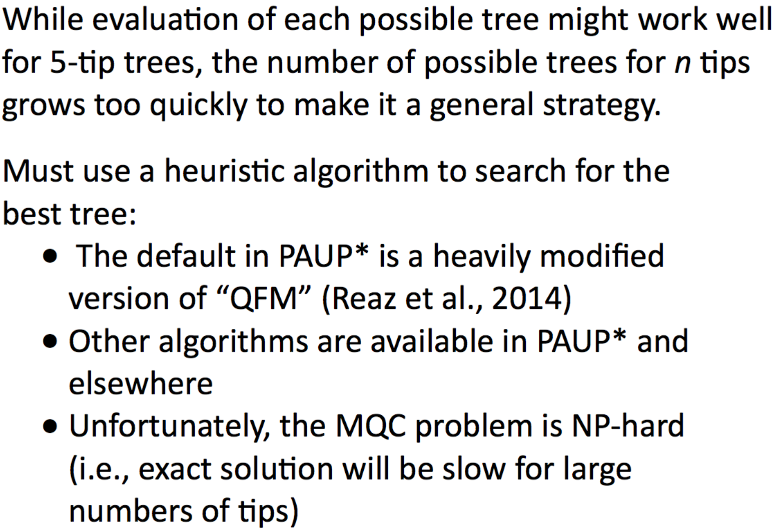
12|34
12|35
12|45
14|35
23|45



2 consistent quartets, 3 inconsistent quartet

Now evaluate the remaining 13 trees and choose the one that maximizes the number of consistent quartets

More than 4 taxa



While evaluation of each possible tree might work well for 5-tip trees, the number of possible trees for *n* tips grows too quickly to make it a general strategy.

Must use a heuristic algorithm to search for the best tree:

- The default in PAUP* is a heavily modified version of “QFM” (Reaz et al., 2014)
- Other algorithms are available in PAUP* and elsewhere
- Unfortunately, the MQC problem is NP-hard (i.e., exact solution will be slow for large numbers of tips)

On to the tutorial!