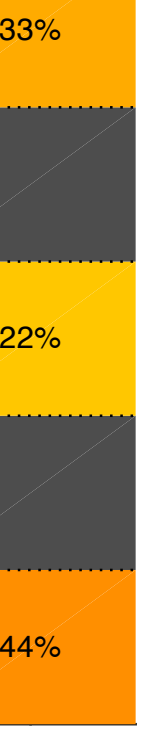
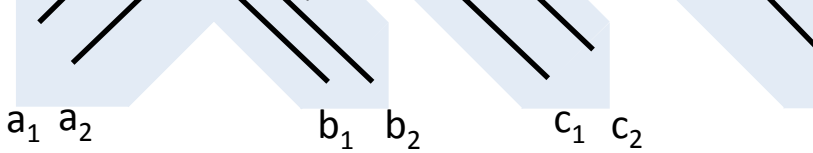
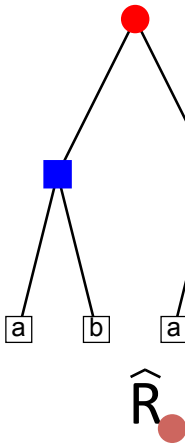




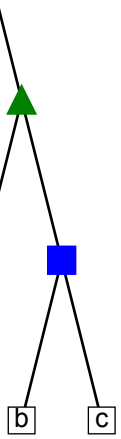
Graph-based Adjustment of Orthology-Relations



Paul Klemm

University of Greifswald, Germany

32th TBI Winterseminar, Bled 2017



$$\prod_{i=1}^{n-1} \binom{numST_i + (a_i - 1)}{a_i}$$

ERNST MORITZ ARNDT
UNIVERSITÄT GREIFSWALD

Wissen lockt. Seit 1456

Outline

Introduction

Phylogenetic, Orthology relation	Cographs	Graph-based orthology inference

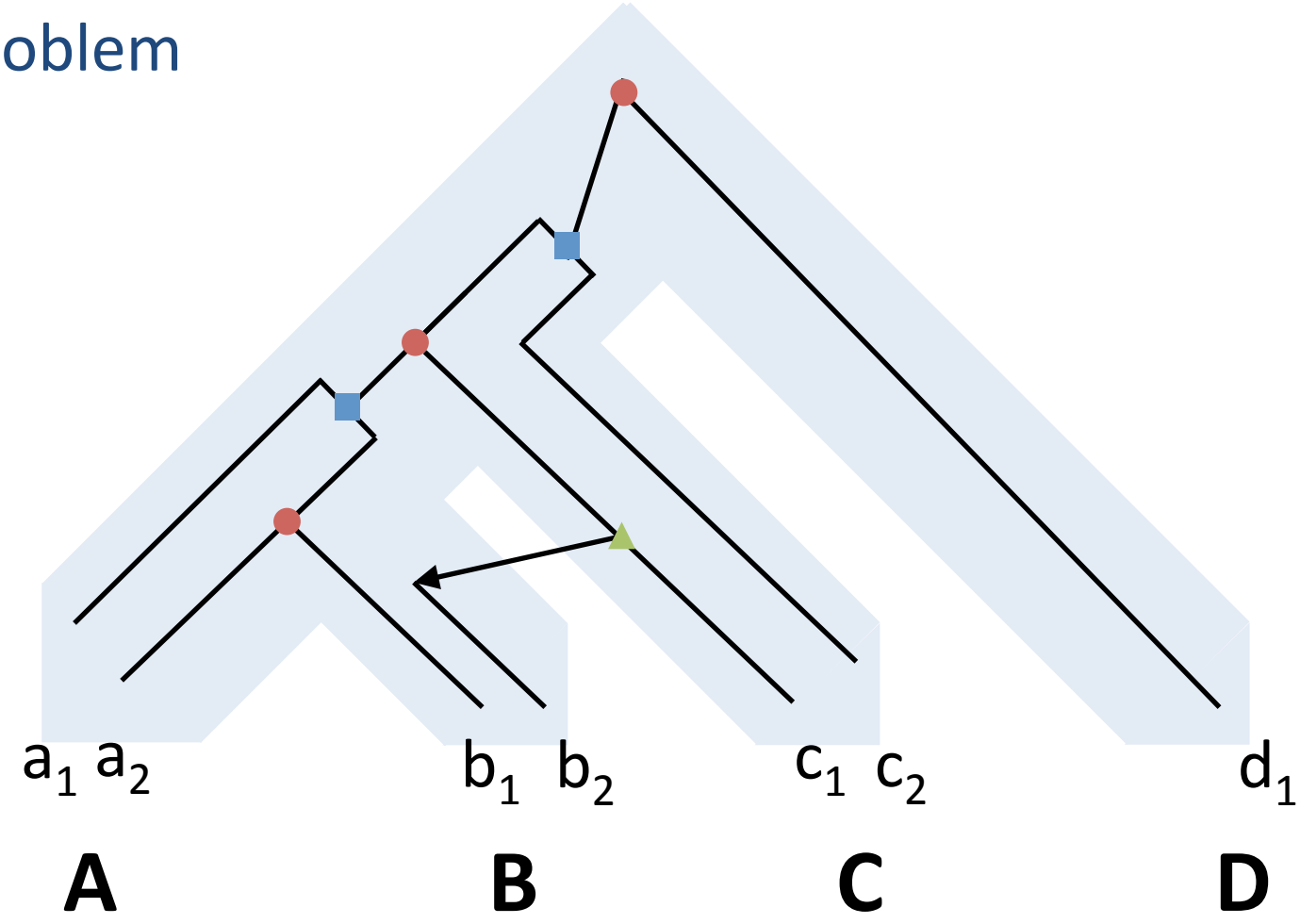
Artificial Data Analysis

Enumeration of all gene trees	Results

Summary and Outlook

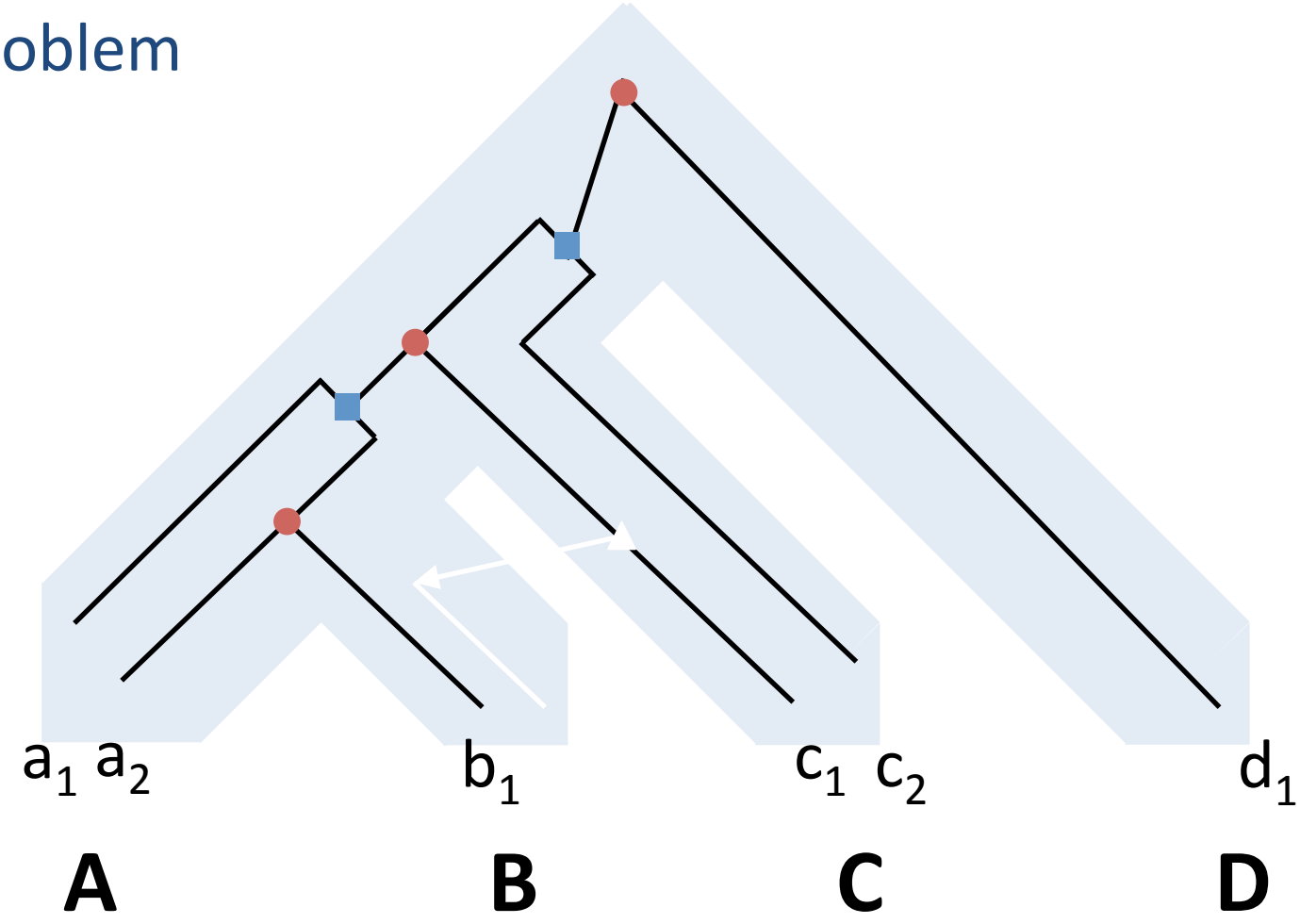
Introduction

The problem



Introduction

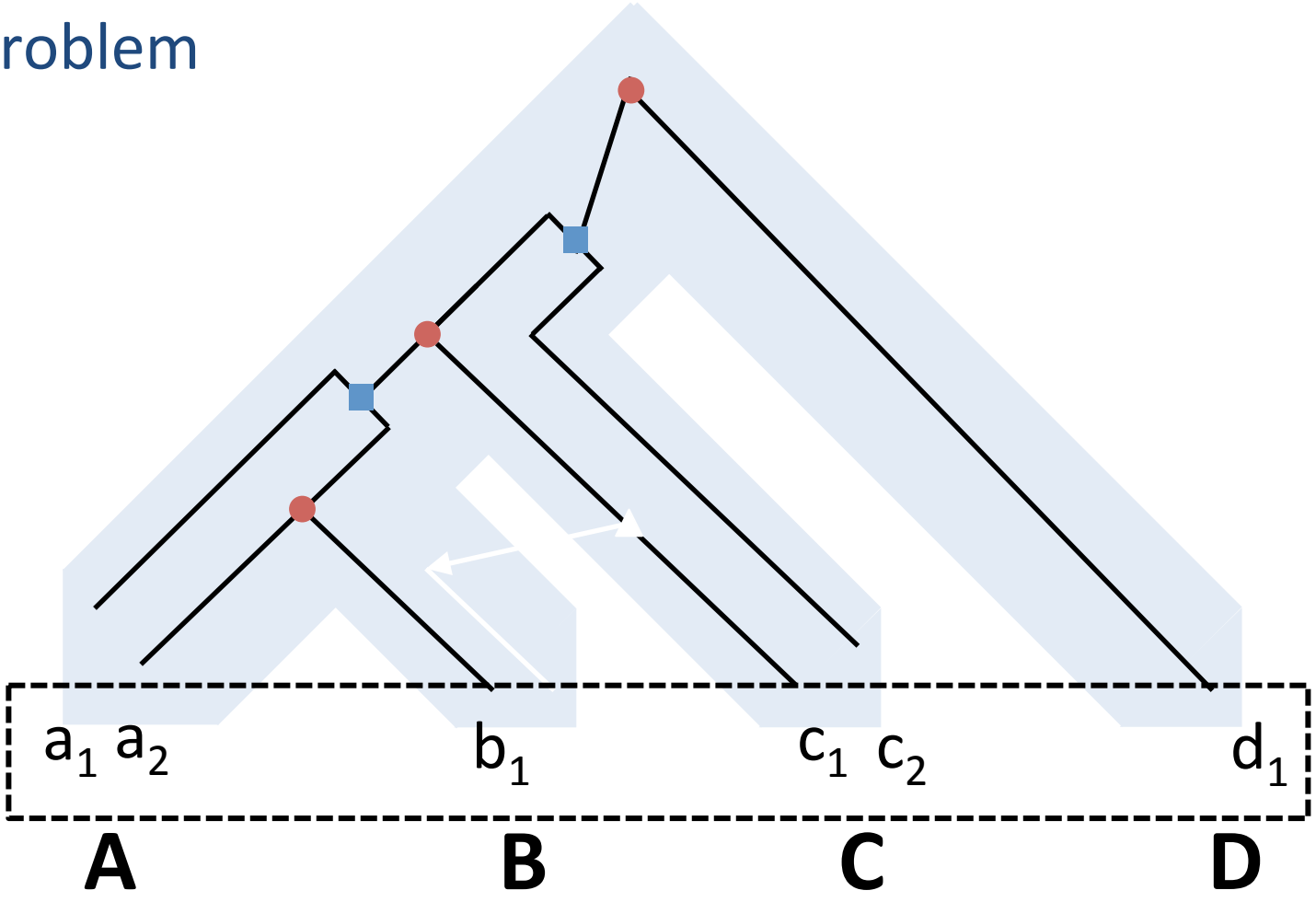
The problem



→ Simplification: no HGT-event

Introduction

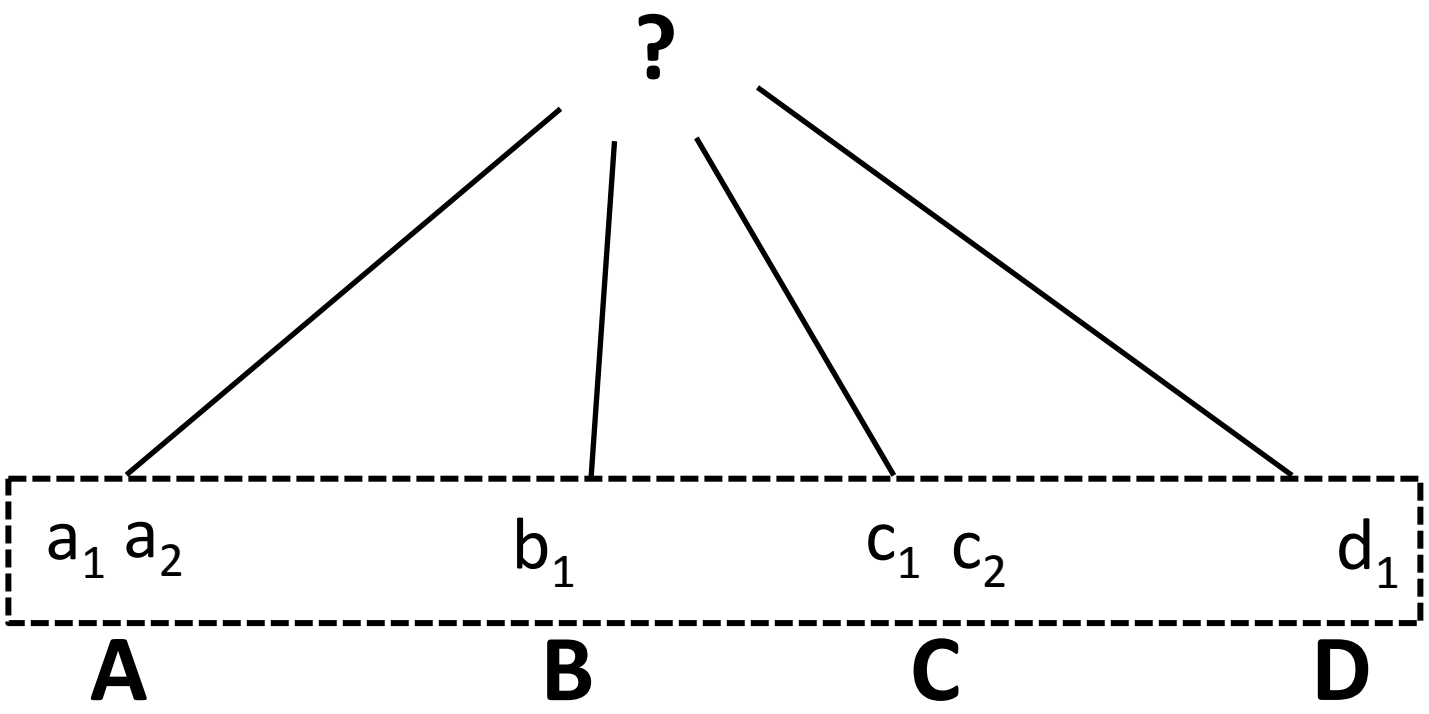
The problem



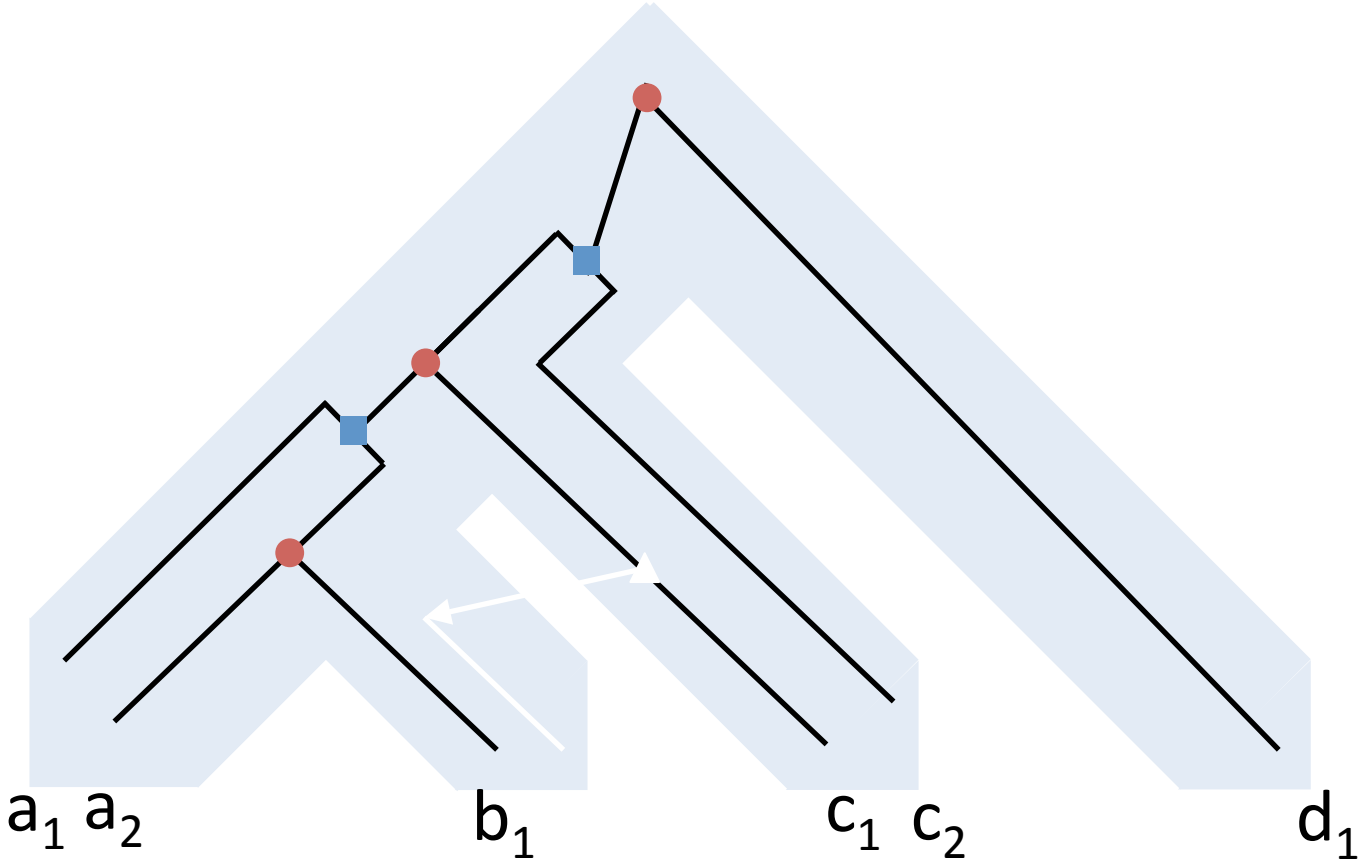
→ Simplification: no HGT-event

Introduction

The problem



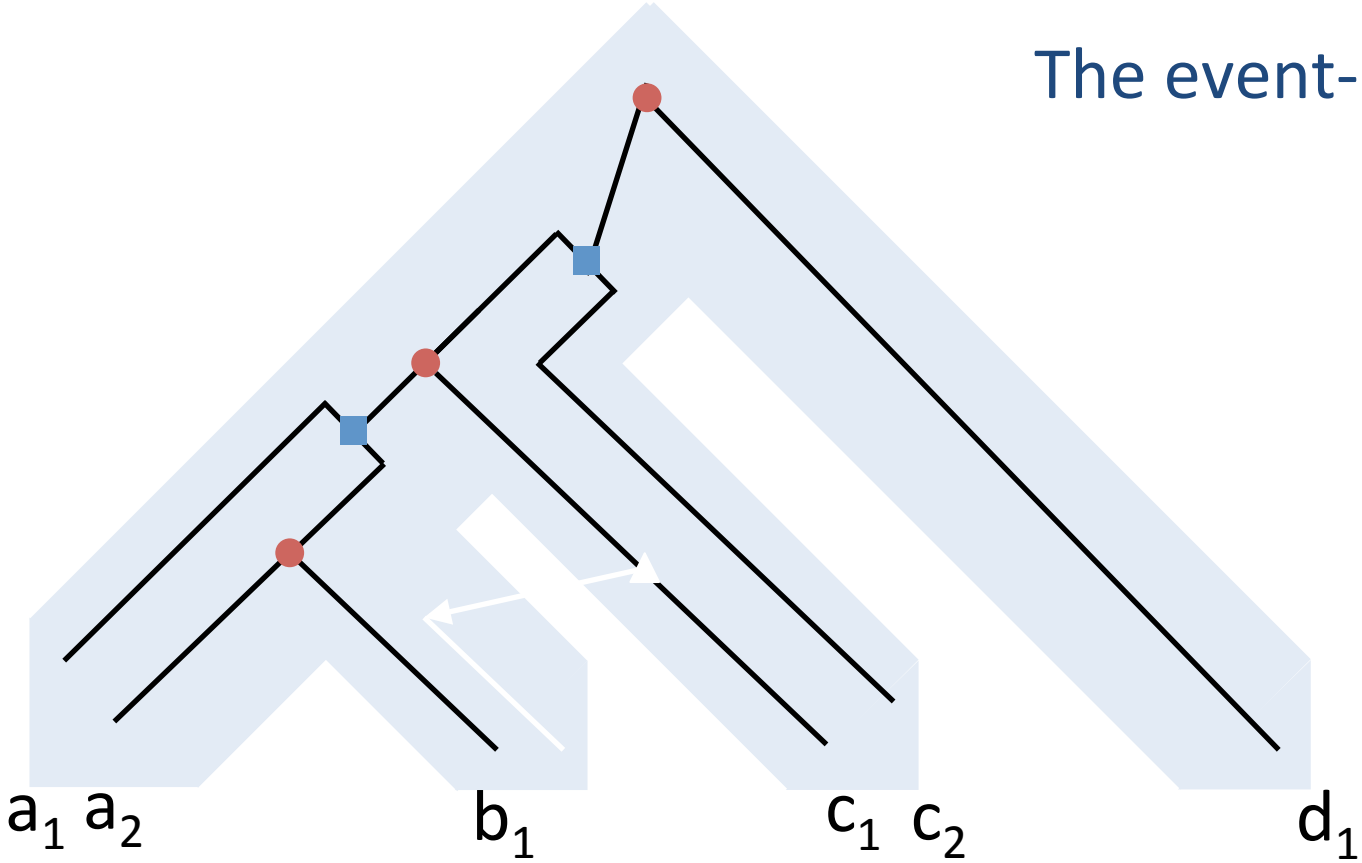
→ Simplification: no HGT-event



We call two genes x and y

- ortholog if lca(x,y) = ● (speciation) e.g. a_2 and b_1
- paralog if lca(x,y) = ■ (duplication) e.g. c_1 and c_2
- xenolog if lca(x,y) = ▲ (HGT) e.g. b_2 and c_1

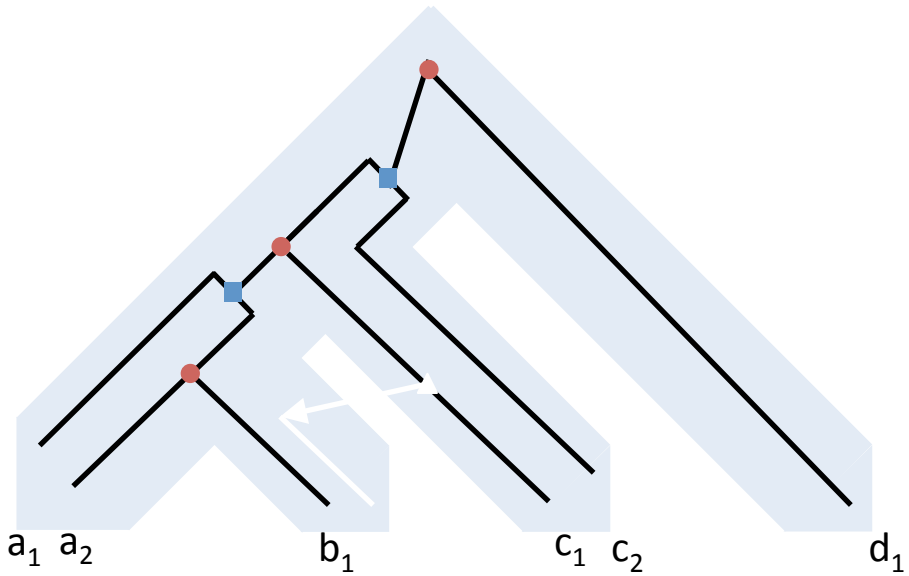
The event-Relations



The (distinct, binary, symmetric) **event-Relations**:

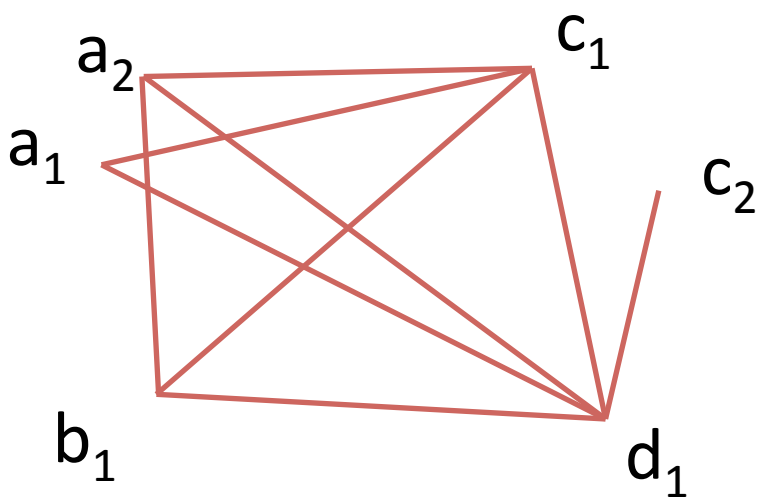
- $R_{●}$ = the set of all (x,y) with lca(x,y) = ● (speciation)
- $R_{■}$ = ... with lca(x,y) = ■ (duplication)
- $R_{▲}$ = ... with lca(x,y) = ▲ (HGT)

The event-Relations



Simplification: no HGT-events

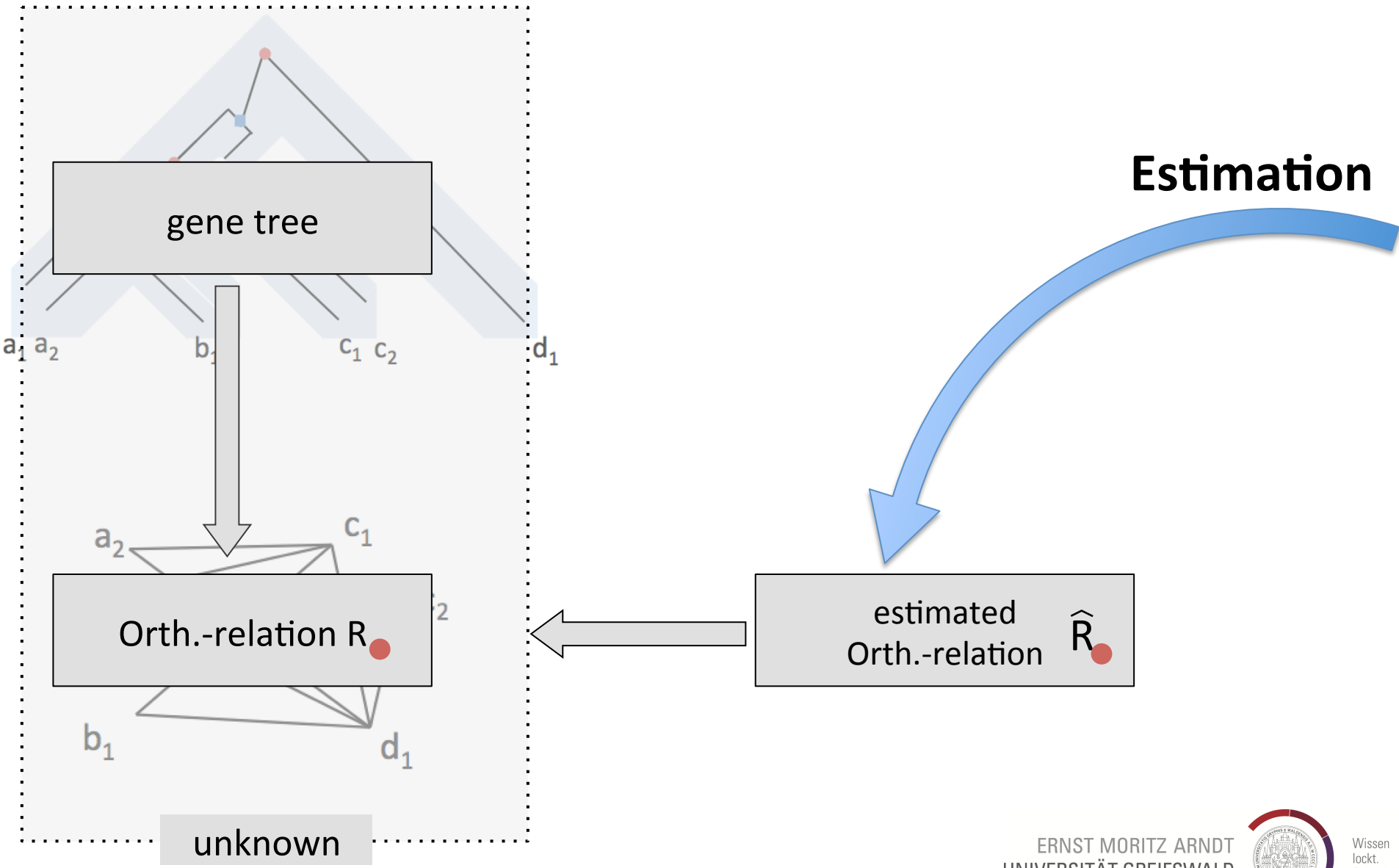
→ $R_{●} = \overline{R_{■}}$



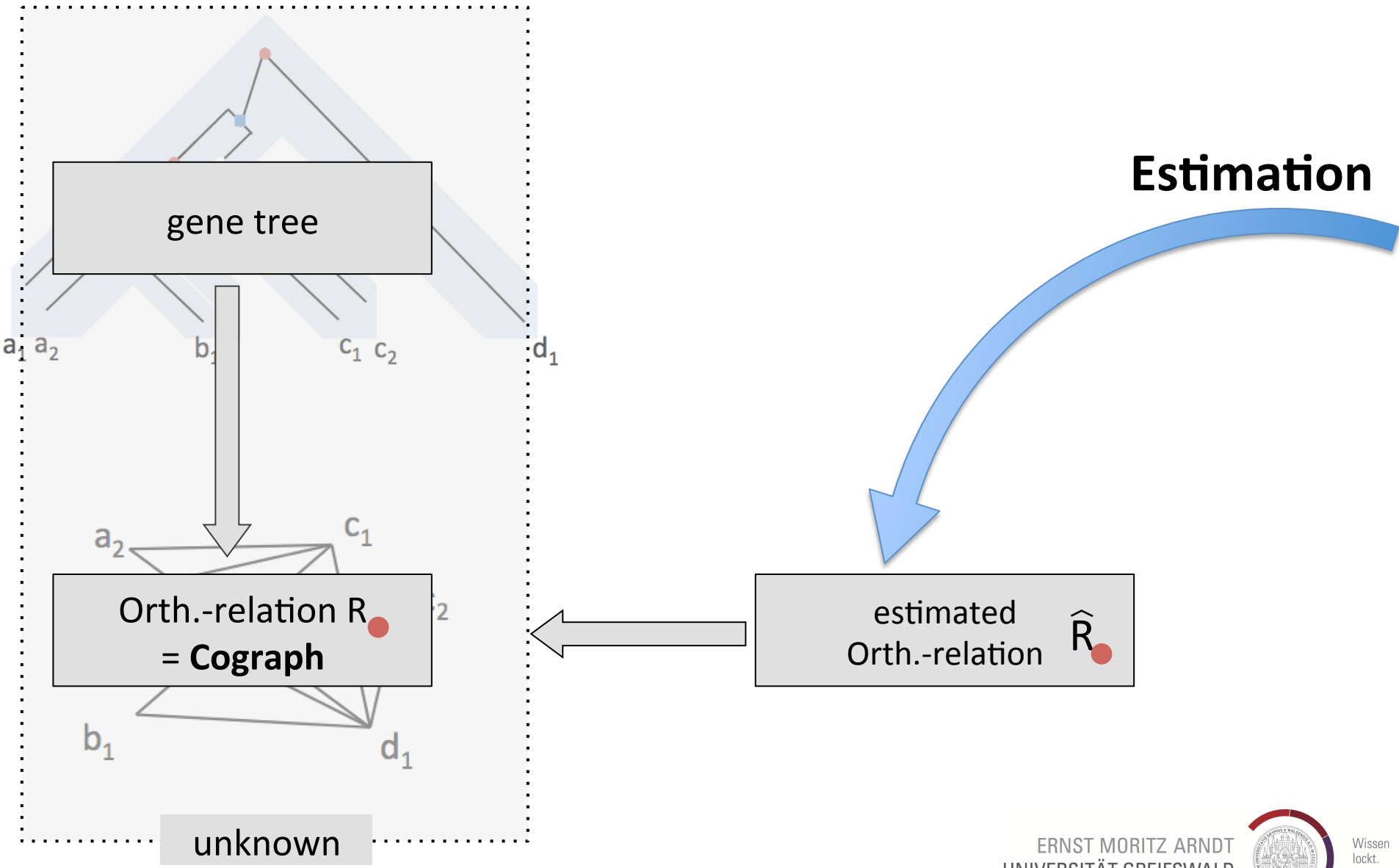
→ graph representation of

$R_{●}$ and $R_{■}$

Introduction



Introduction

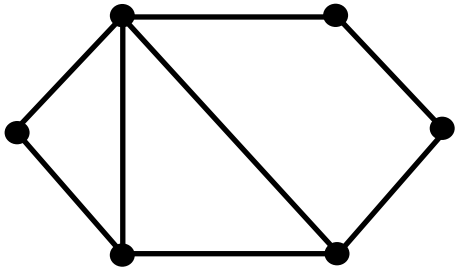


Cographs

A graph is called **cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

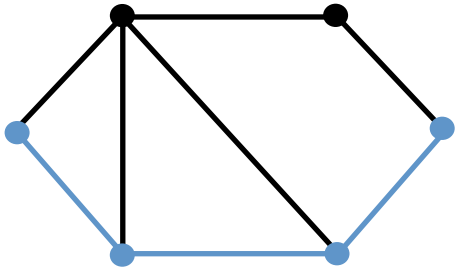
Cographs

A graph is called **cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()



Cographs

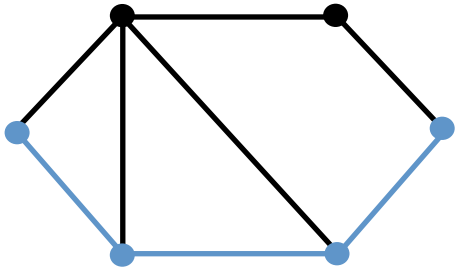
A graph is called **cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()



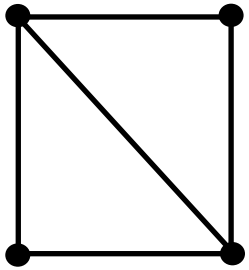
not a Cograph

Cographs

A graph is called **cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

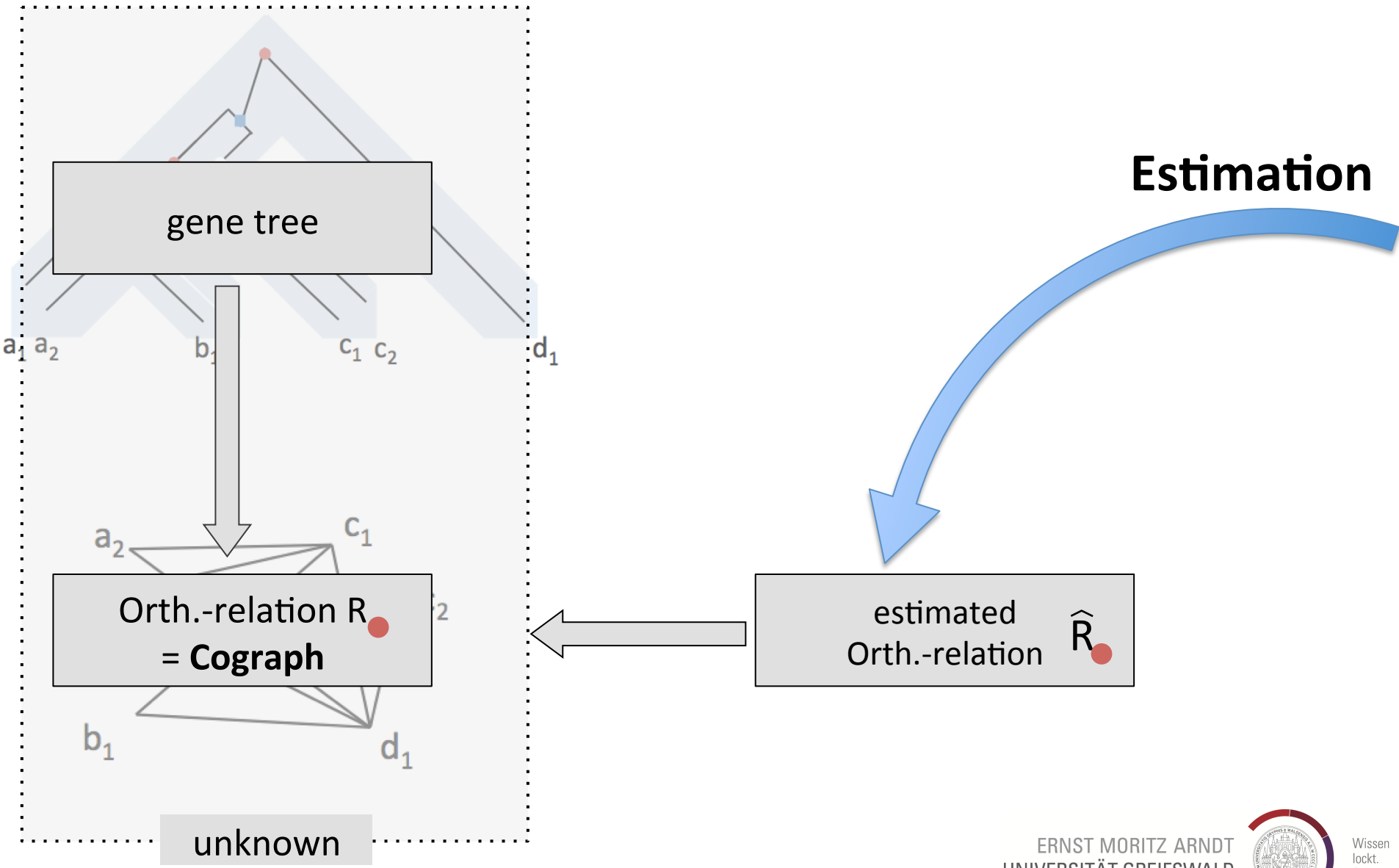


not a Cograph

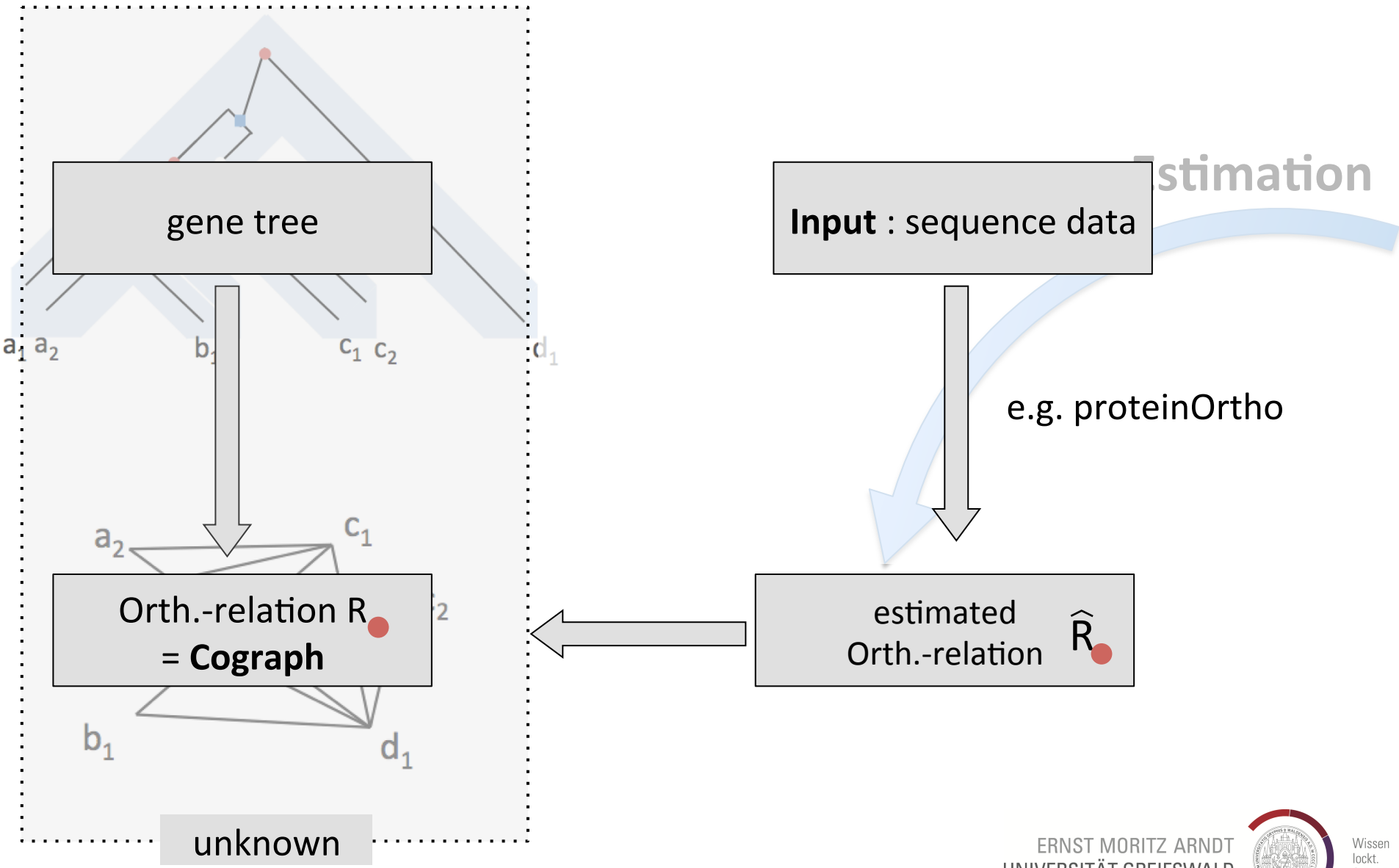


Cograph

Introduction



Introduction



Introduction

Graph-based orthology inference (e.g. proteinOrtho)

Input : sequence data

Local alignment search
(e.g. BLAST)

Sequence similarity **s(x,y)**

Introduction

Graph-based orthology inference (e.g. proteinOrtho)

Input : sequence data

Local alignment search
(e.g. BLAST)

Sequence similarity **s(x,y)**

Then two genes x in X and y in Y are **estimated orthologs** if:

i. they are from different species and
ii. **s(x,y)** is the ~best score compared to **s(x‘,y)** and **s(x,y‘)** where x‘ in X, y‘ in Y

Graph-based orthology inference (e.g. proteinOrtho)

Then two genes x and y are **estimated orthologs** if:

i. they are from different species and
ii. **s(x,y)** is the ~best score compared to **s(x‘,y)** and **s(x,y‘)** where x‘ in X, y‘ in Y

Simplification: leaf distance d(x,y) of the „true“ gene tree:

d(x,y) = length of the shortest path between x and y

Graph-based orthology inference (e.g. proteinOrtho)

Then two genes x and y are **estimated orthologs** if:

i. they are from different species and
ii. **s(x,y)** is the ~best score compared to **s(x‘,y)** and **s(x,y‘)** where x‘ in X, y‘ in Y

Simplification: leaf distance d(x,y) of the „true“ gene tree:

d(x,y) = length of the shortest path between x and y

$$d \sim 1/s$$

Graph-based orthology inference (e.g. proteinOrtho)

Then two genes x and y are **estimated orthologs** if:

i. they are from different species and
ii. **s(x,y)** is the ~best score compared to **s(x‘,y)** and **s(x,y‘)** where x‘ in X, y‘ in Y

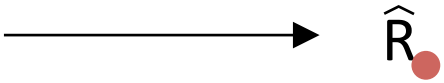
Simplification: leaf distance d(x,y) of the „true“ gene tree:

d(x,y) = length of the shortest path between x and y

$d \sim 1/s$

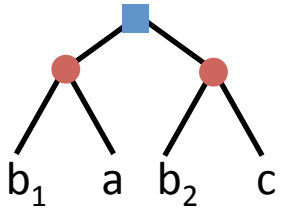
→ **estimated orthologs** if:

i. they are from different species and
ii. the distance d(x,y) is „small“



Introduction

Graph-based orthology inference



Genes: b_1 a b_2 c

Species: B A B C

$R_{\bullet}$:

$\hat{R}_{\bullet}$:

Simplification: leaf distance d(x,y) of the „true“ gene tree:

d(x,y) = length of the shortest path between x and y

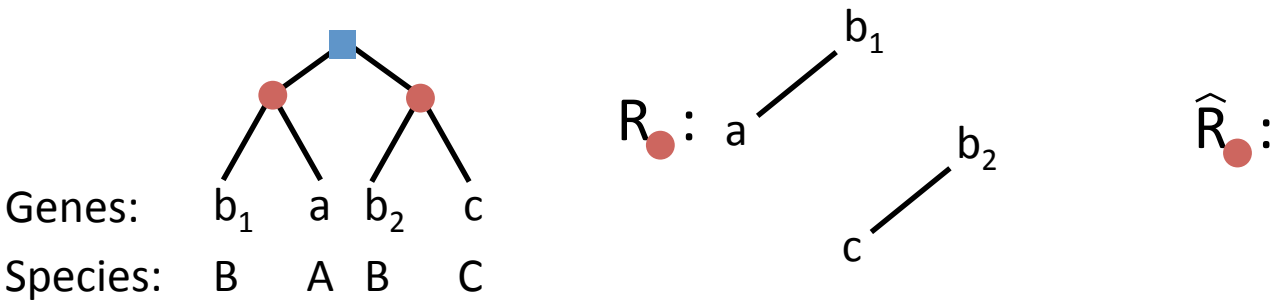
→ **estimated orthologs** if:

i. they are from different species and
ii. the distance d(x,y) is „small“

→ $\hat{R}_{\bullet}$

Introduction

Graph-based orthology inference

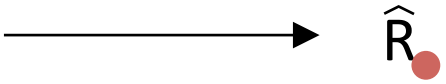


Simplification: leaf distance d(x,y) of the „true“ gene tree:

d(x,y) = length of the shortest path between x and y

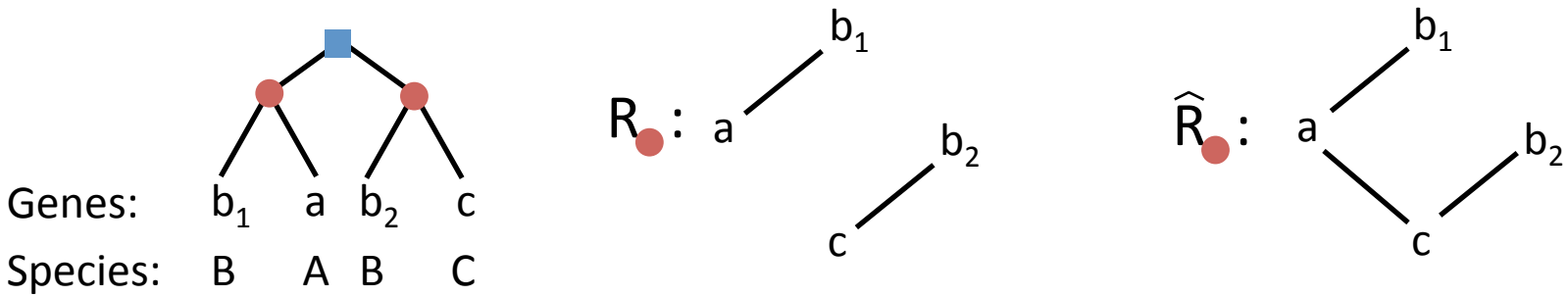
→ **estimated orthologs** if:

i. they are from different species and
ii. the distance d(x,y) is „small“



Introduction

Graph-based orthology inference

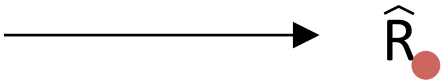


Simplification: leaf distance d(x,y) of the „true“ gene tree:

d(x,y) = length of the shortest path between x and y

→ **estimated orthologs** if:

i. they are from different species and
ii. the distance d(x,y) is „small“



Artificial Data Analysis

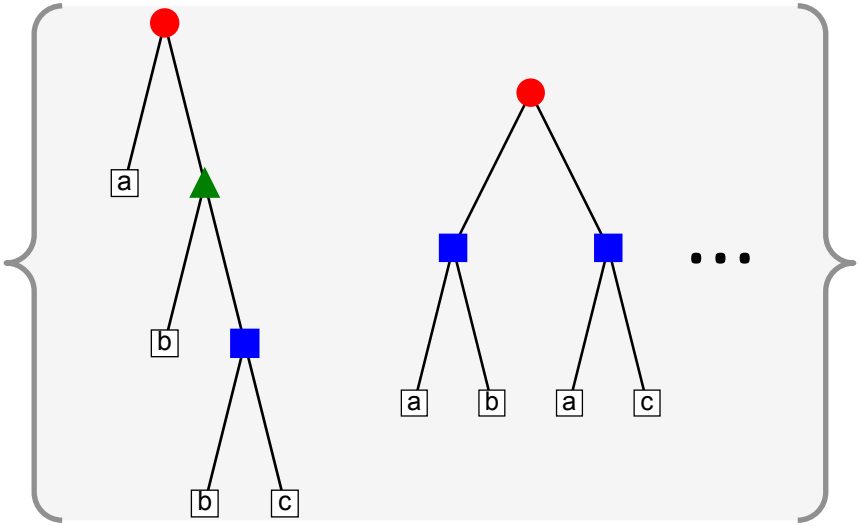
Question:

How much information can we infer from non-cograph relations ?

How much information can we infer from non-cograph relations ?

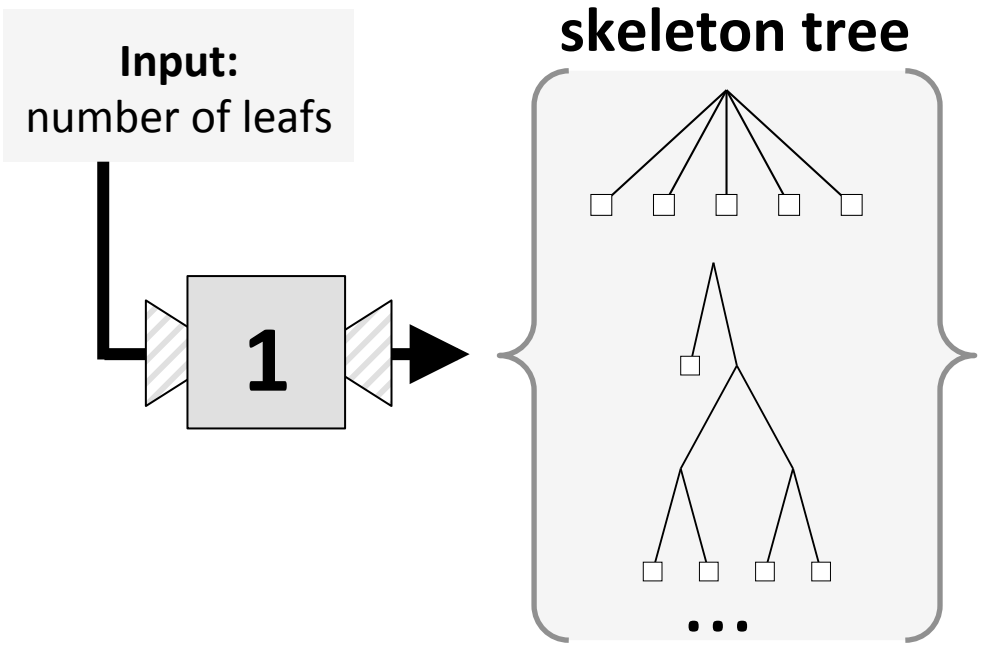
Input:
number of leafs

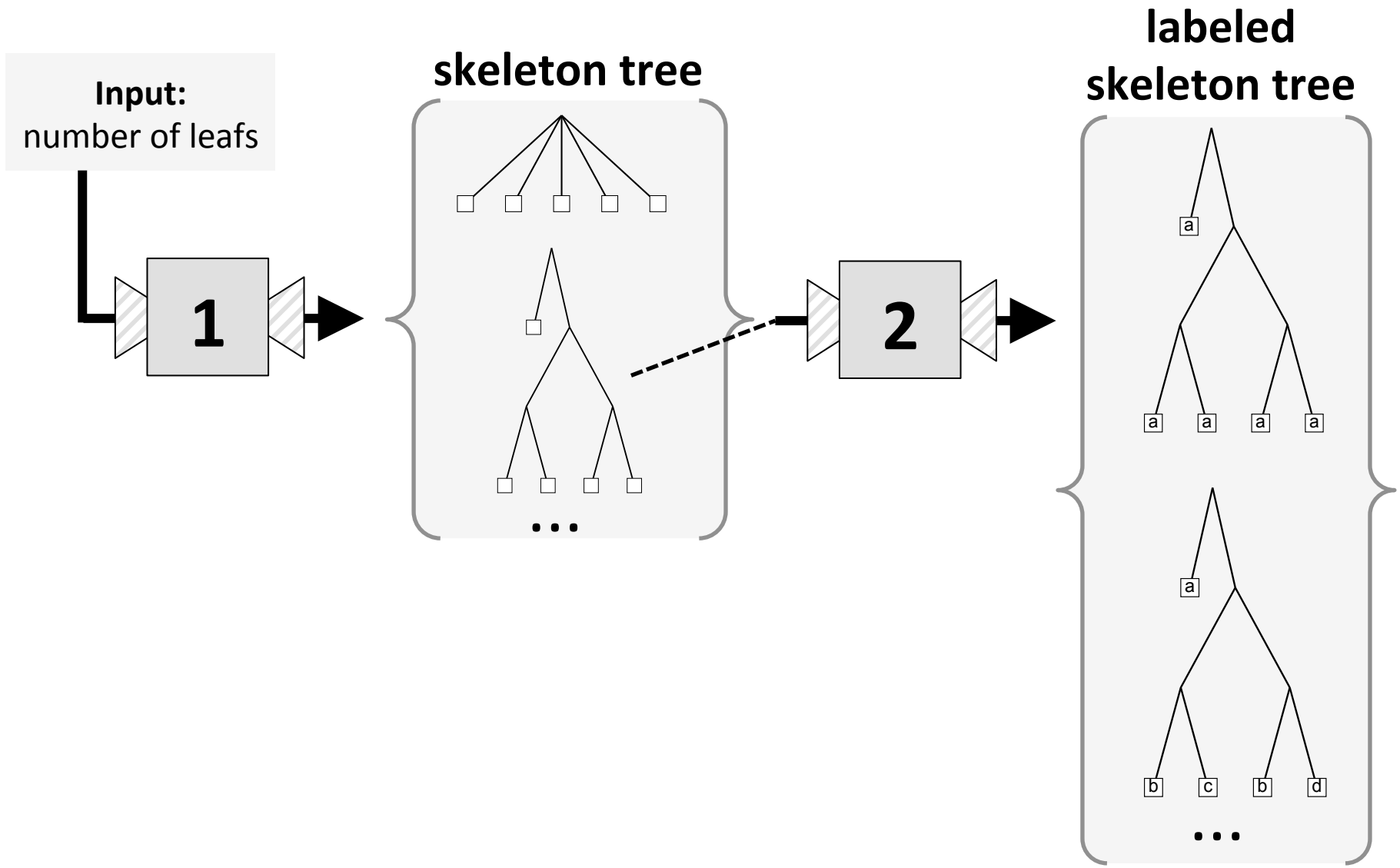
1,2,3

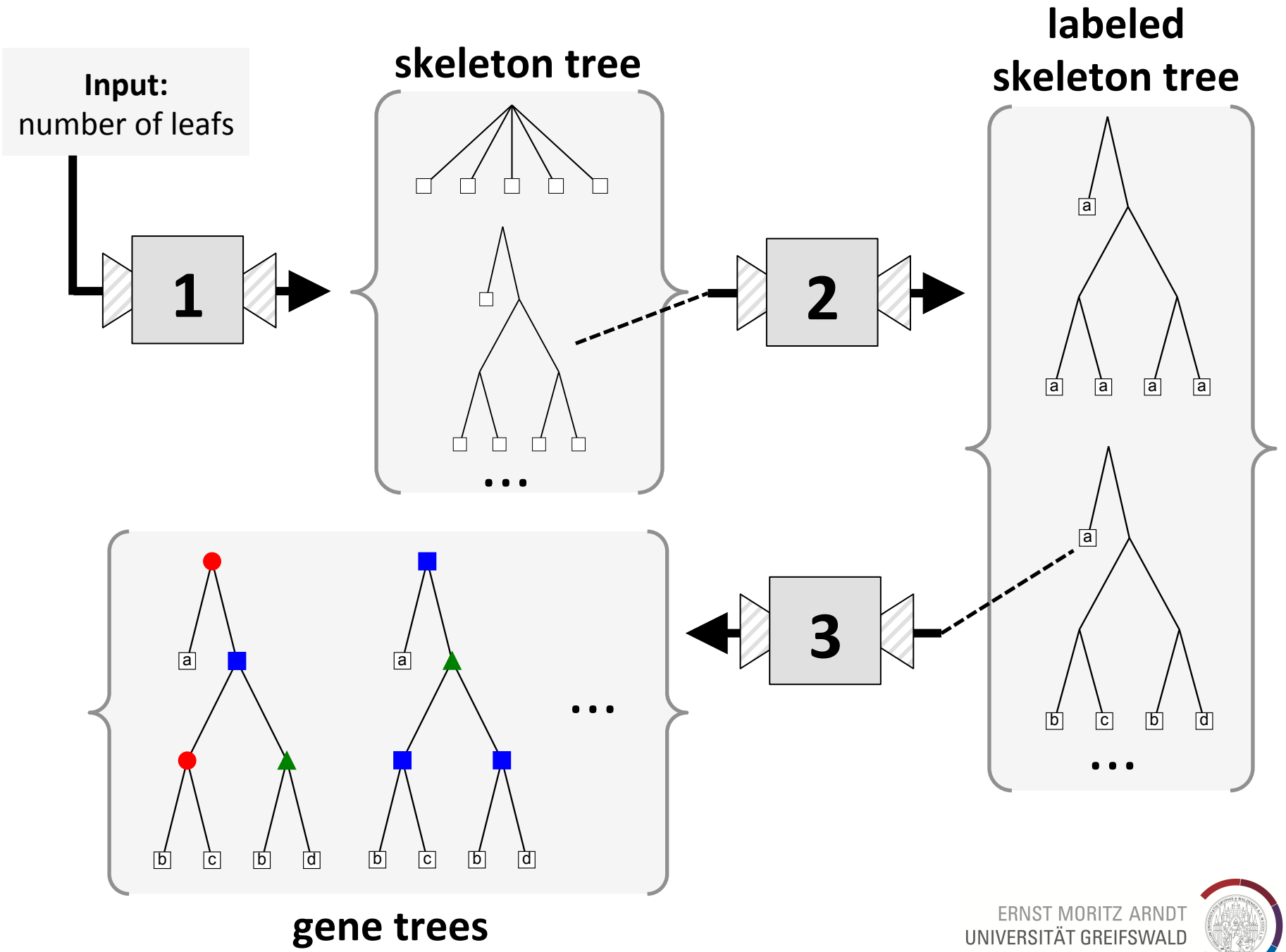


gene trees

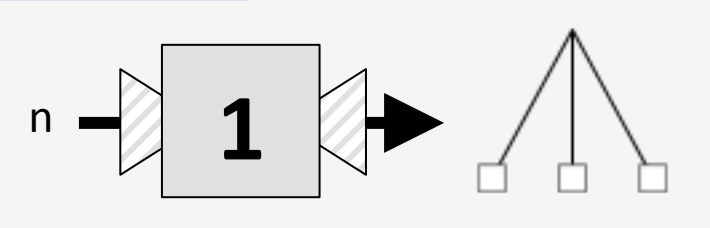
estimated Orth.-relation $\hat{R}$ + Orth.-relation R


Input:
number of leafs
1
skeleton tree
...


Input:
number of leafs
1
skeleton tree
2
labeled
skeleton tree
a
a
a
a
a
a
b
c
b
d


Input:
number of leafs
1
skeleton tree
2
labeled
skeleton tree
3
gene trees

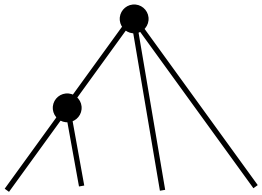
Artificial Data Analysis



1. Generate skeleton trees

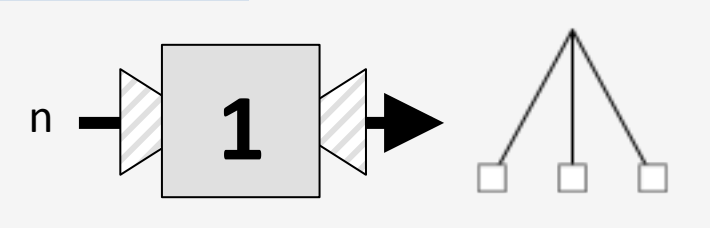
We want:

1. Rooted trees with n leafs



1 + 1 + 1 + 1 = 4

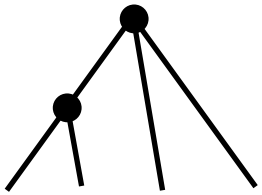
Artificial Data Analysis



1. Generate skeleton trees

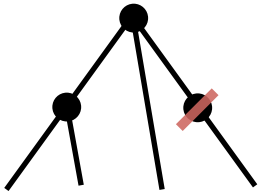
We want:

1. Rooted trees with n leafs

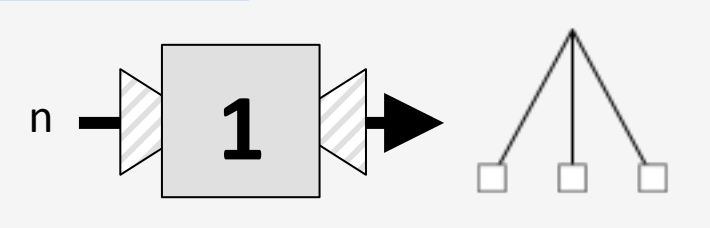


1 + 1 + 1 + 1 = 4

2. All inner nodes should have at least 2 childs



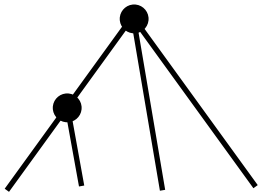
Artificial Data Analysis



1. Generate skeleton trees

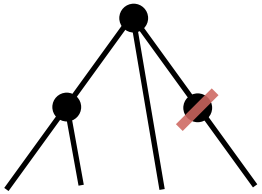
We want:

1. Rooted trees with n leafs

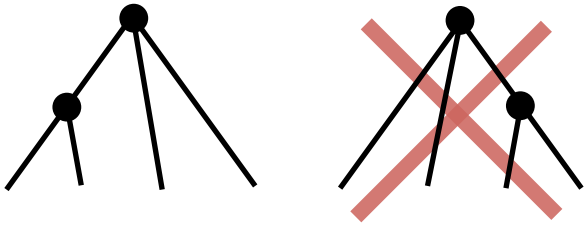


1 + 1 + 1 + 1 = 4

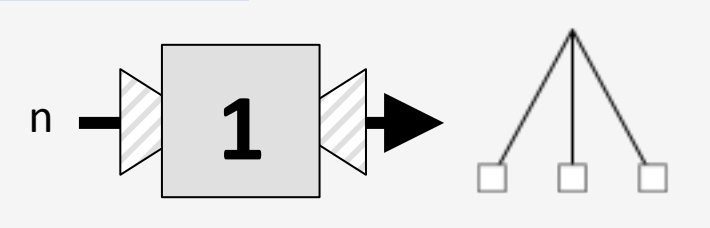
2. All inner nodes should have at least 2 childs



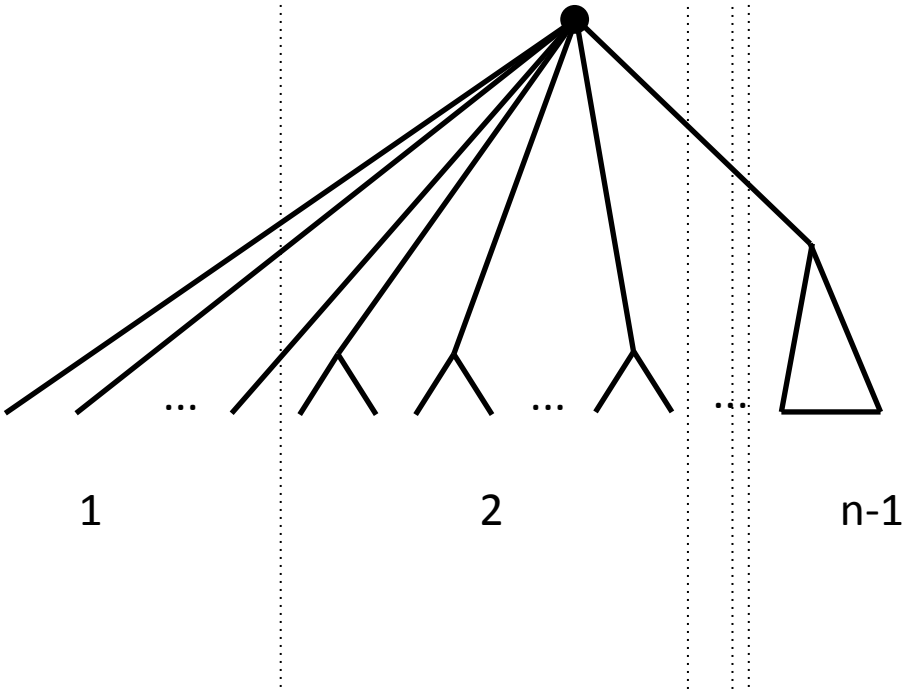
3. Only one representative of the isomorphism classes



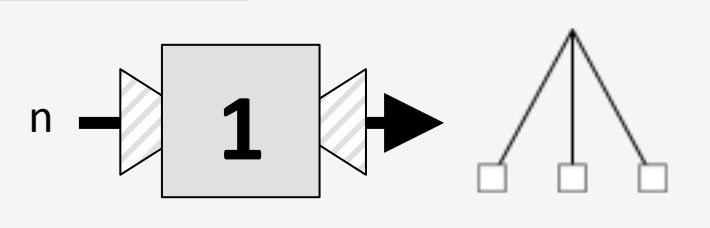
Artificial Data Analysis



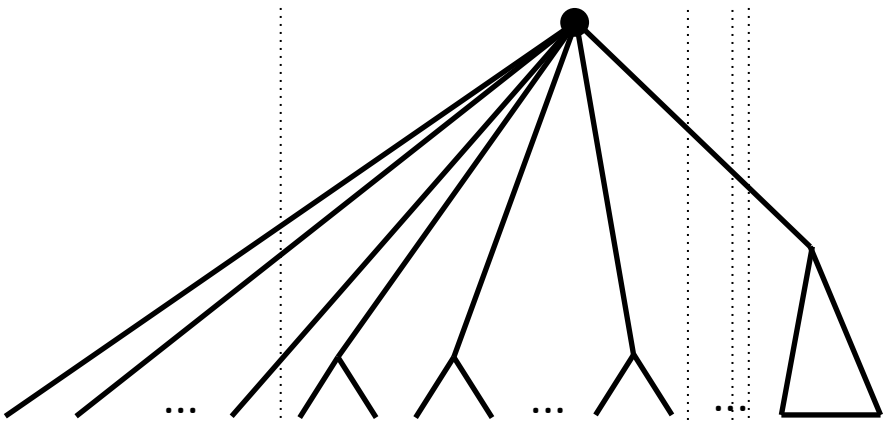
1. Generate skeleton trees



Artificial Data Analysis



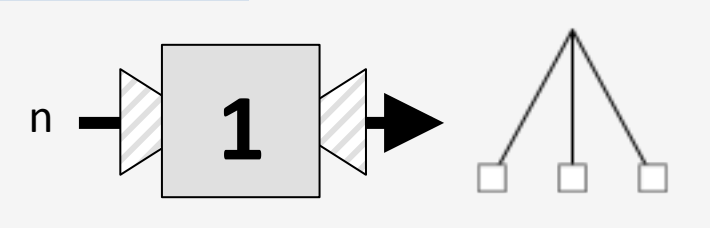
1. Generate skeleton trees



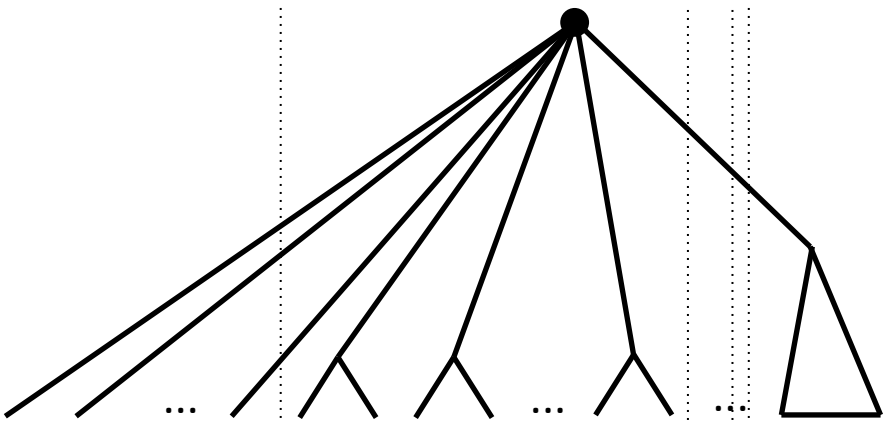
1 2 n-1

$1 \cdot a_1 + 2 \cdot a_2 + \ldots + (n-1) \cdot a_{n-1} = n$

Artificial Data Analysis



1. Generate skeleton trees



1 2 n-1

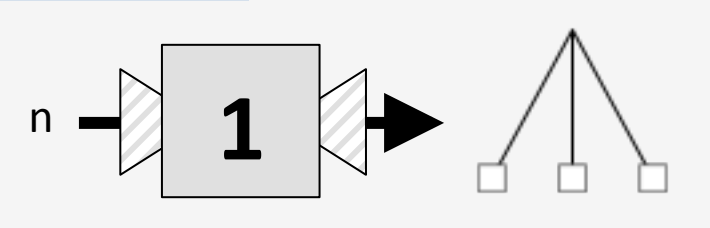
$1 \cdot a_1 + 2 \cdot a_2 + \ldots + (n-1) \cdot a_{n-1} = n$

Partition problem:

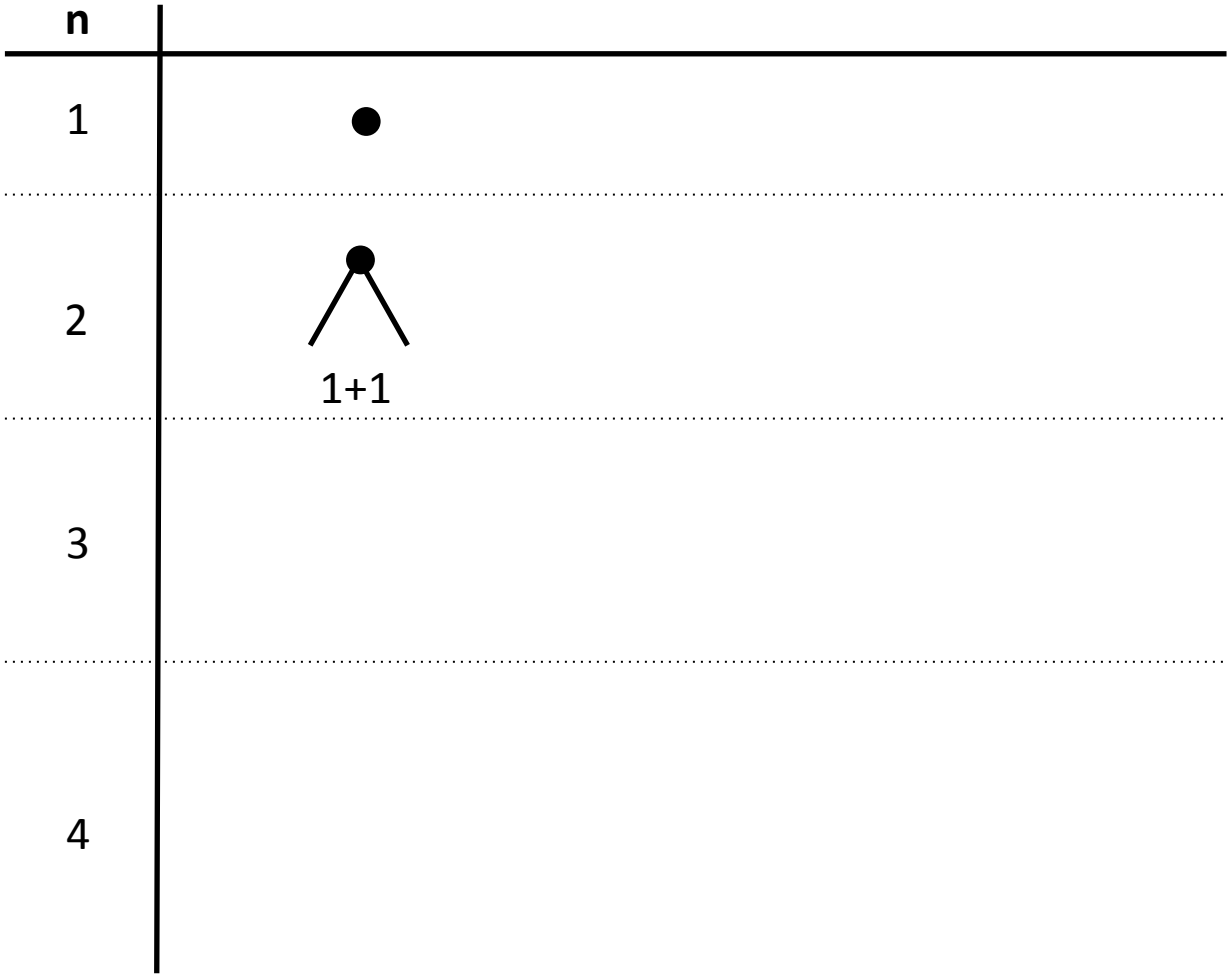
given an integer n

Question : Obtain all possible ways to write n as sum of 1 , ... , n-1

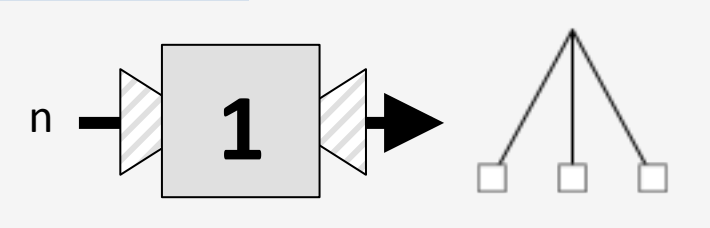
Artificial Data Analysis



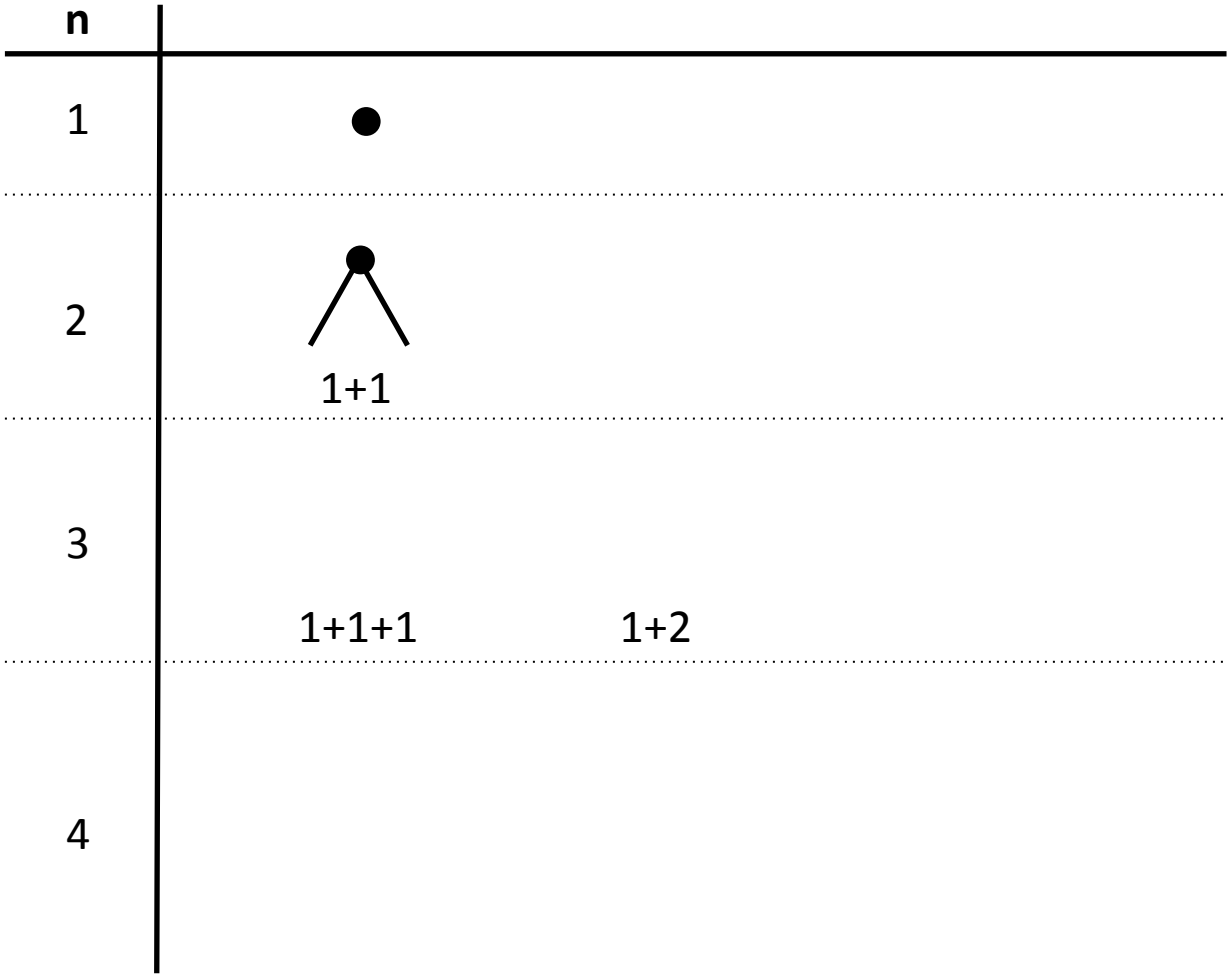
1. Generate skeleton trees



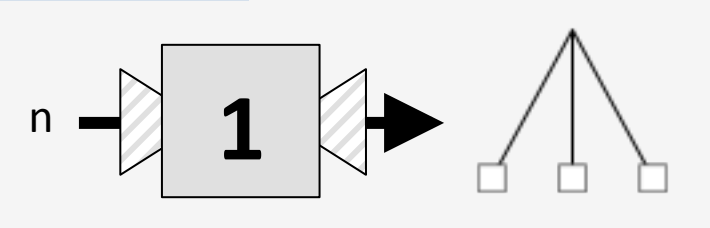
Artificial Data Analysis



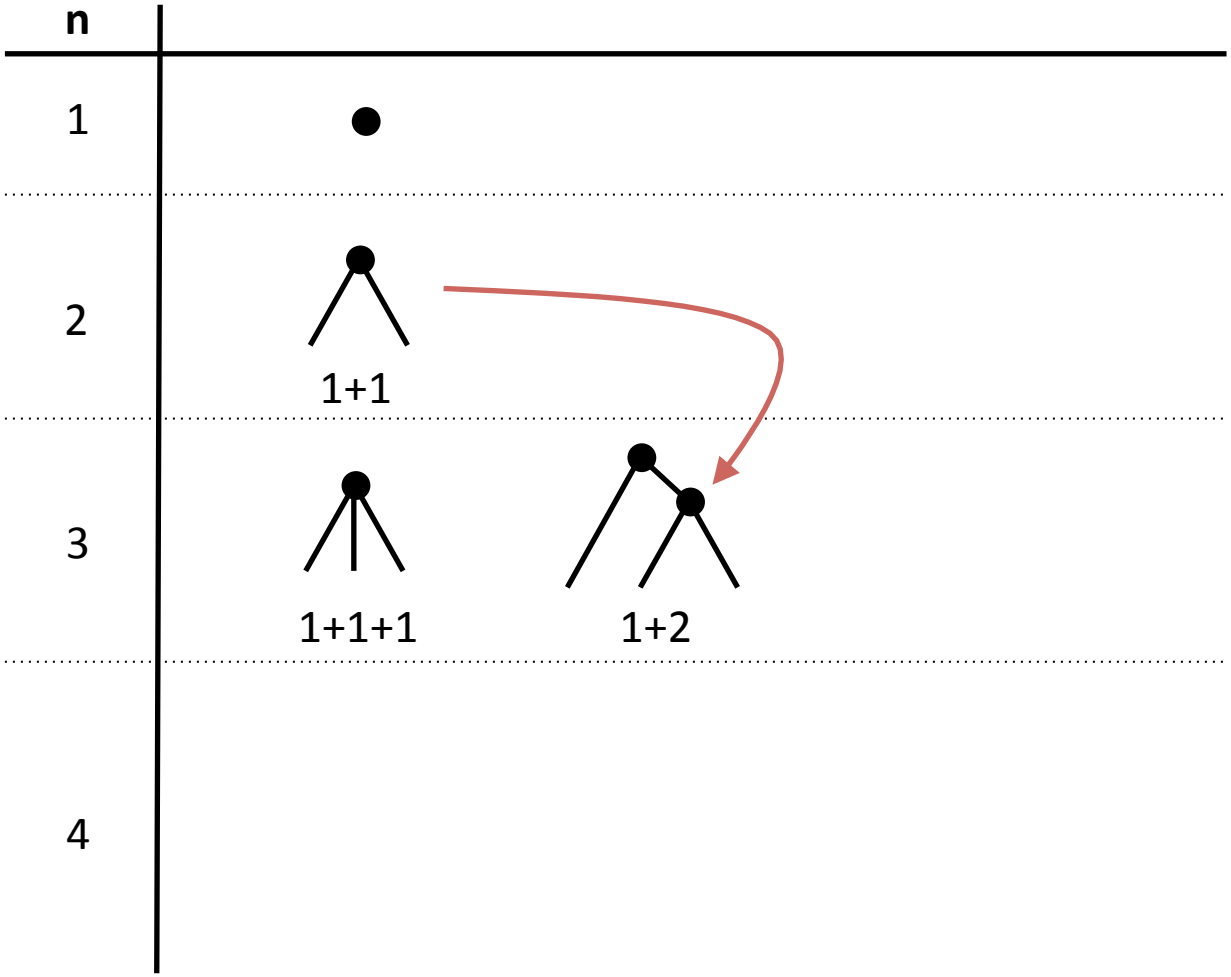
1. Generate skeleton trees



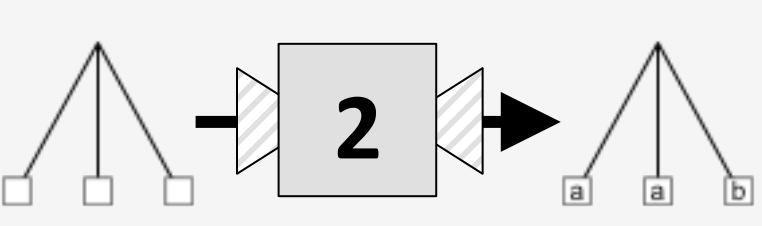
Artificial Data Analysis



1. Generate skeleton trees



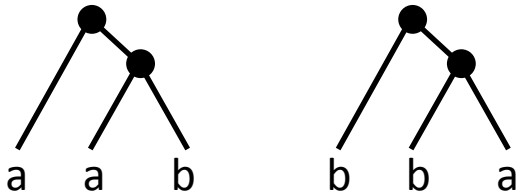
Artificial Data Analysis



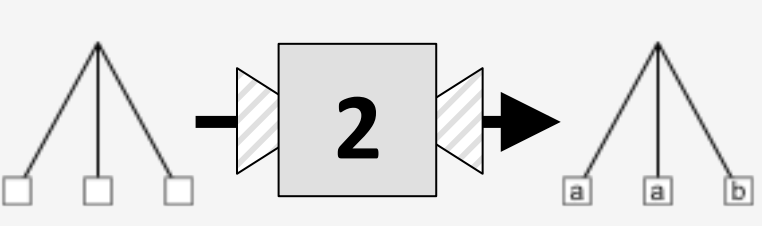
2. Generate labels for skeleton trees

We want:

1. Only one representative over all permutations



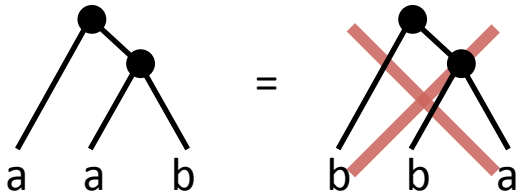
Artificial Data Analysis



2. Generate labels for skeleton trees

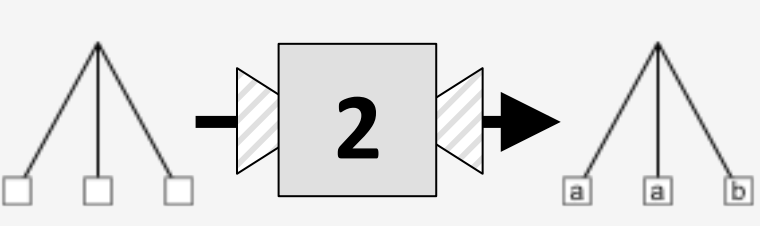
We want:

1. Only one representative over all permutations



{ a → b , b → a }

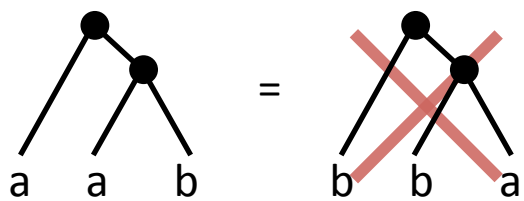
Artificial Data Analysis



2. Generate labels for skeleton trees

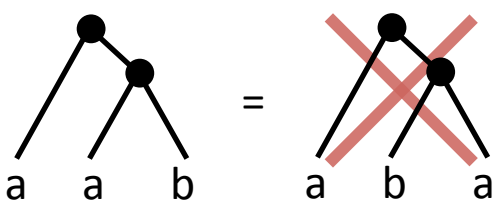
We want:

1. Only one representative over all permutations

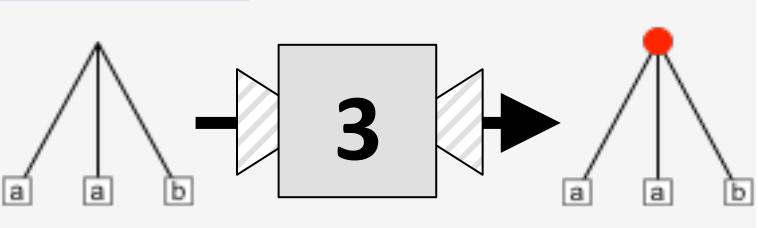


{ a → b , b → a }

2. Only one representative of the isomorphism classes



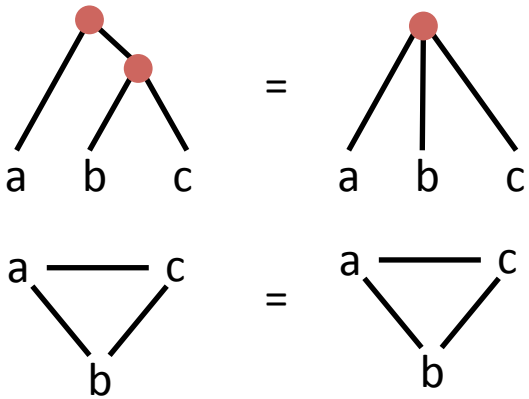
Artificial Data Analysis

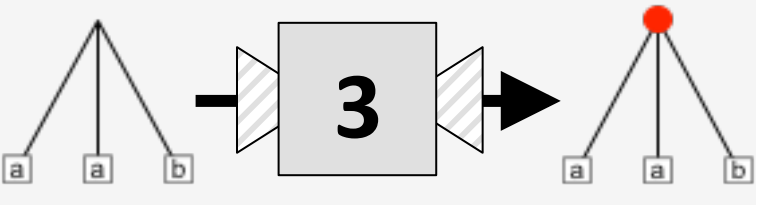


3. Coloring the labeled skeleton trees

We want:

No edge with the same event (color) on both ends

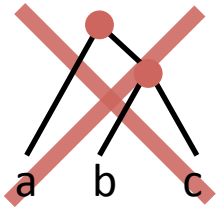




3. Coloring the labeled skeleton trees

We want:

No edge with the same event (color) on both ends

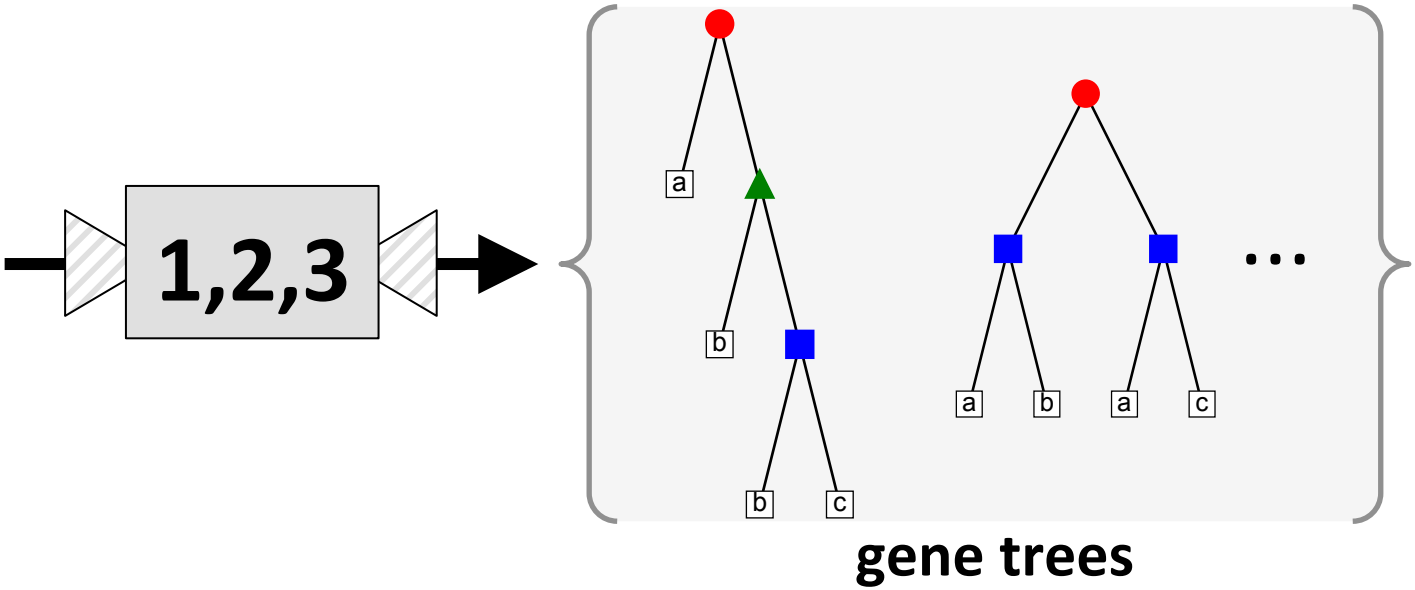


= 3-coloring of the inner nodes

Artificial Data Analysis

Overview

Input:
number of leafs



estimated Orth.-relation $\hat{R}$ + Orth.-relation R

Artificial Data Analysis

Results for n=4 leafs

Total number of gene trees : **324**

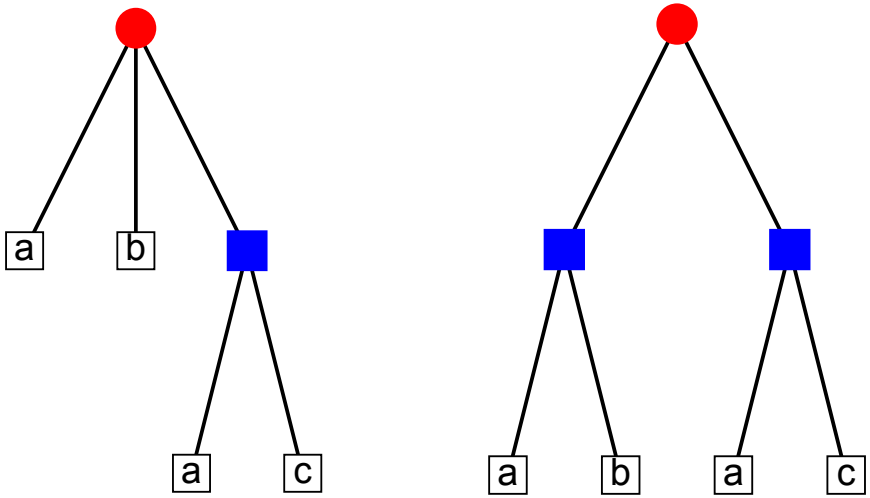
Non-cograph cases : **27**

Artificial Data Analysis

Results for n=4 leafs

Total number of gene trees : **324**

Non-cograph cases : **27**



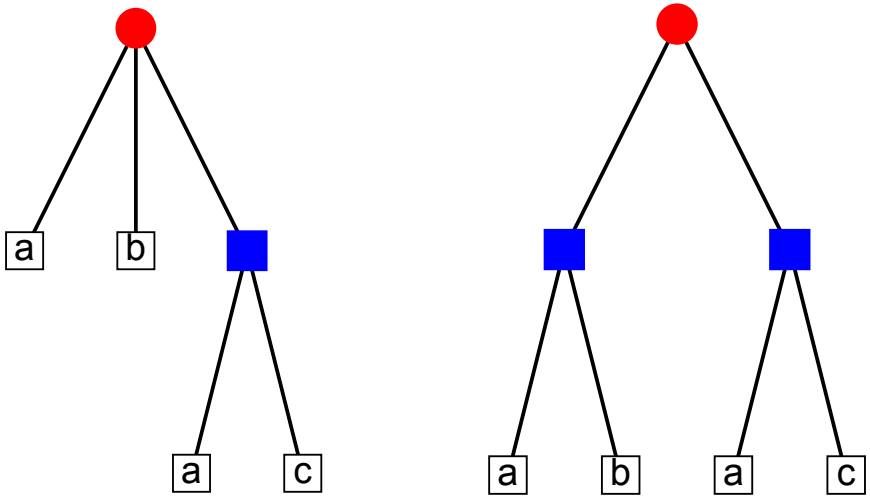
Artificial Data Analysis

Results for n=4 leafs

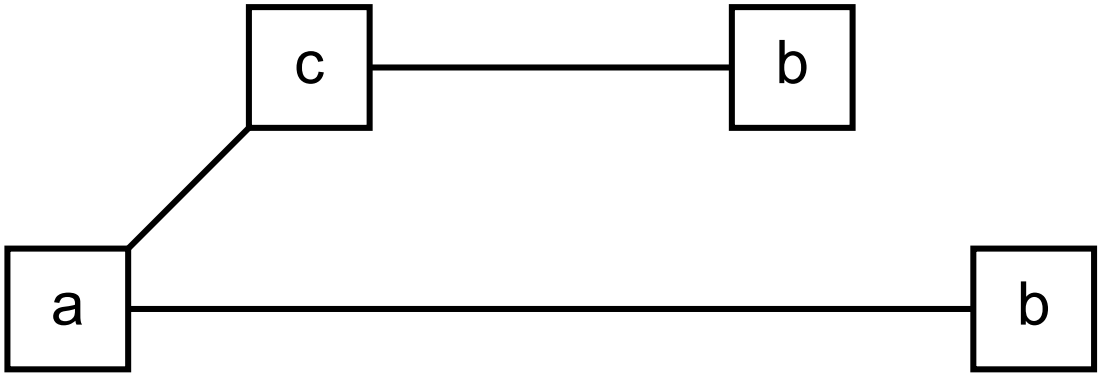
Total number of gene trees : **324**

Non-cograph cases : **27**

Number of problemclasses : **1**



$\widehat{R}$:



problemclass #1

27 generating gene trees

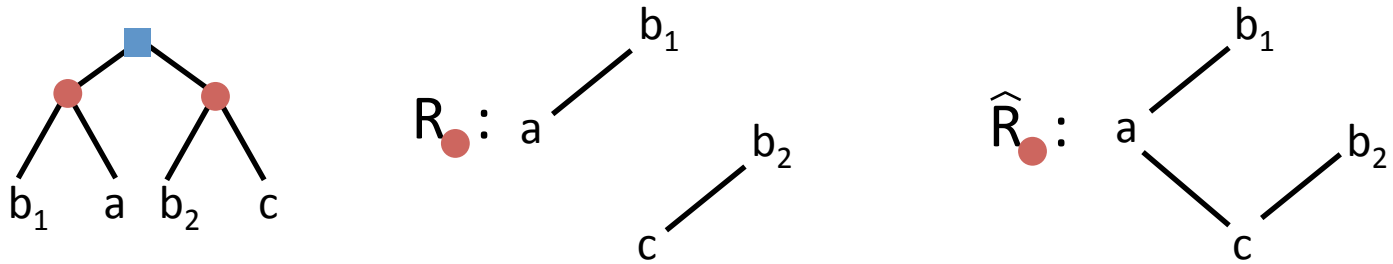
Artificial Data Analysis

Results for n=5 leafs

Total number of gene trees	: **3543**
Non-cograph cases	: **822**
Number of problemclasses	: **9**

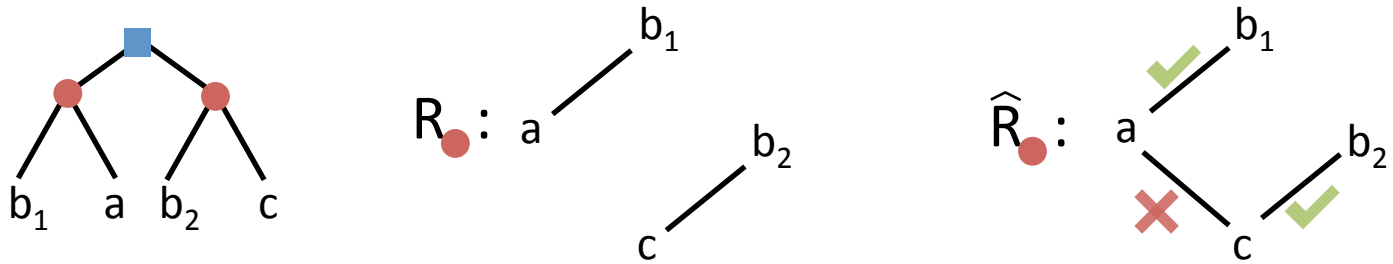
Artificial Data Analysis

Question 1 : Is there a preference of right/wrong edges?



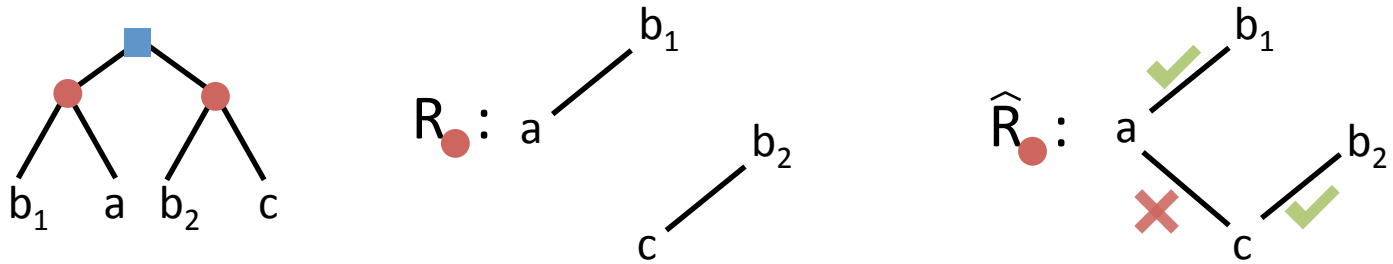
Artificial Data Analysis

Question 1 : Is there a preference of right/wrong edges?

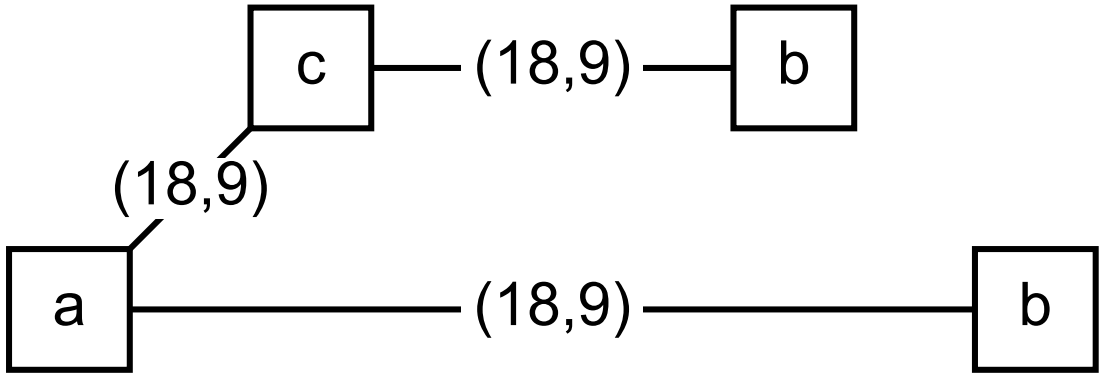


Artificial Data Analysis

Question 1 : Is there a preference of right/wrong edges?



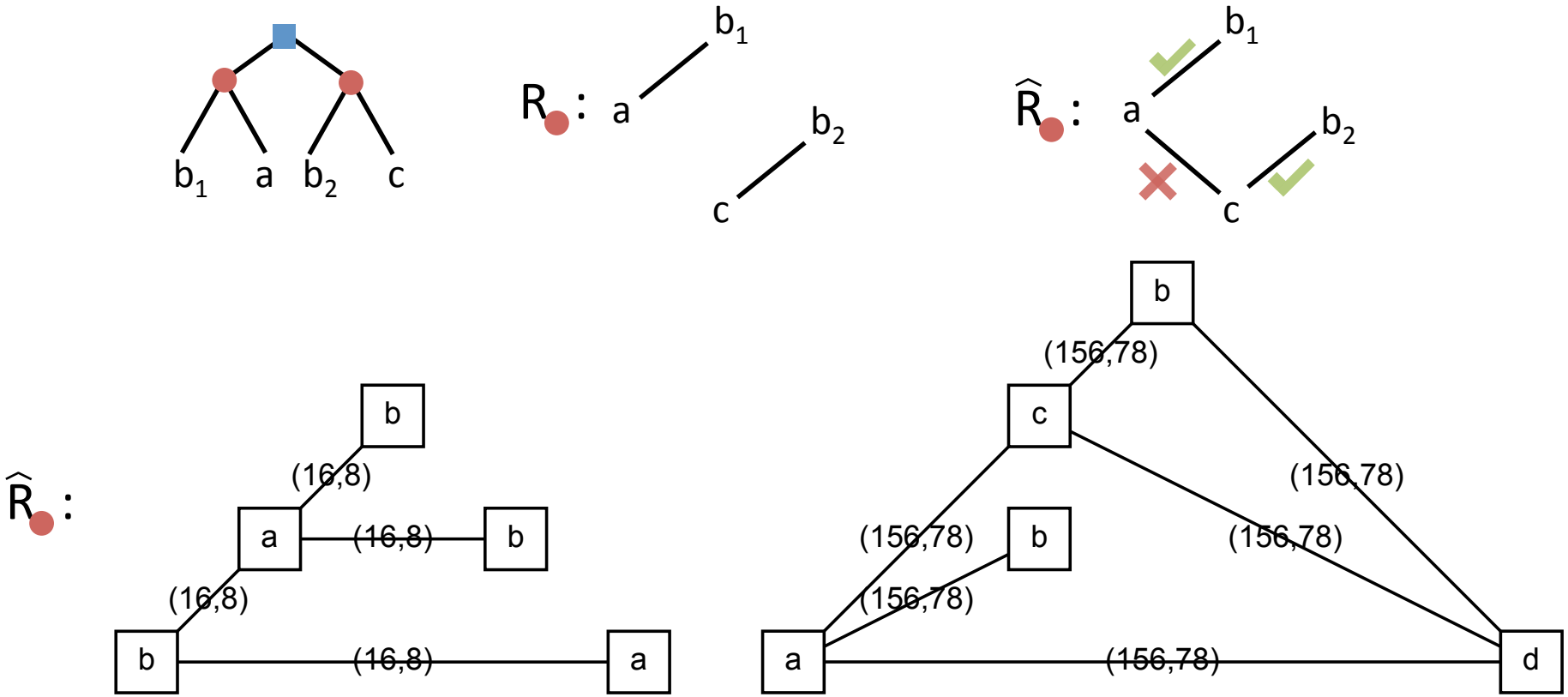
$\hat{R}$:



problemclass #1
27 generating gene trees

Artificial Data Analysis

Question 1 : Is there a preference of right/wrong edges?

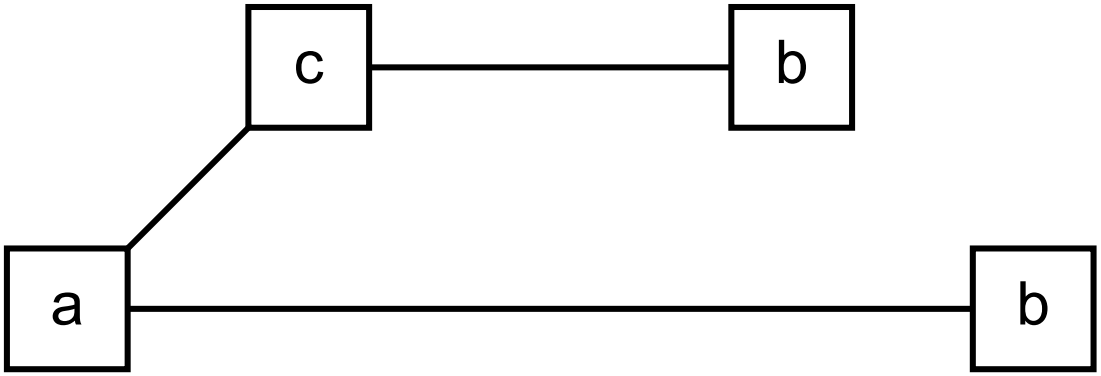


problemclass #3
24 generating gene trees

problemclass #7
234 generating gene trees

Artificial Data Analysis

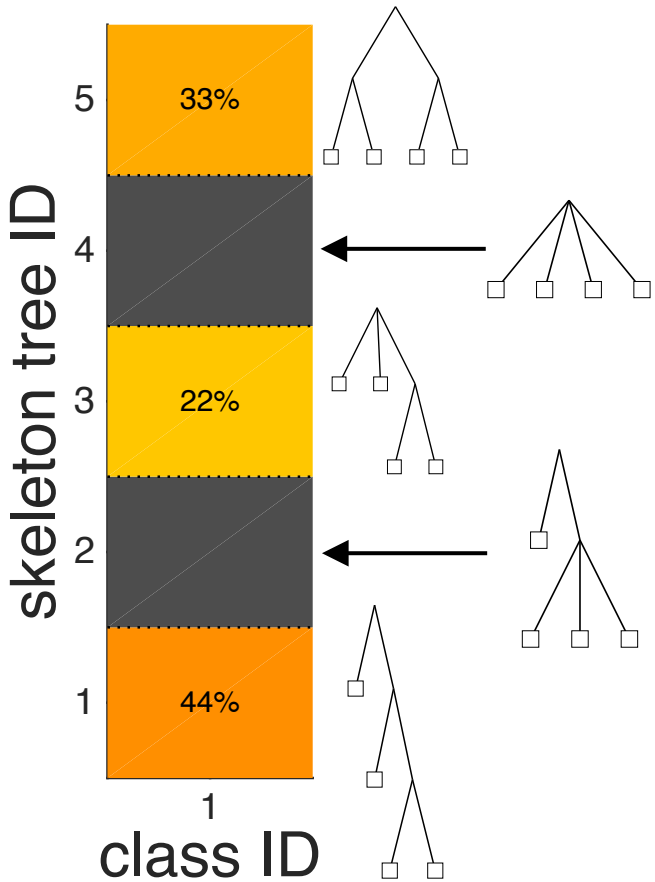
Question 2 : Can one infer information about the skeleton?

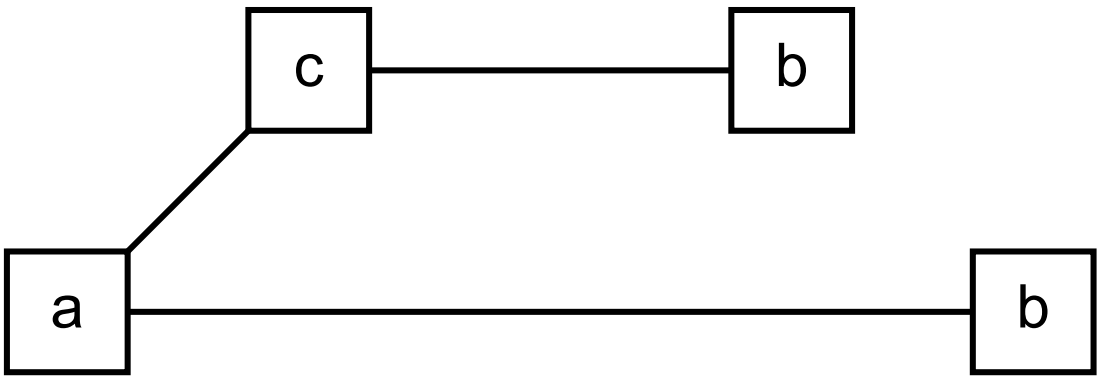


problemclass #1
27 generating gene trees

Artificial Data Analysis

Question 2 : Can one infer information about the skeleton?





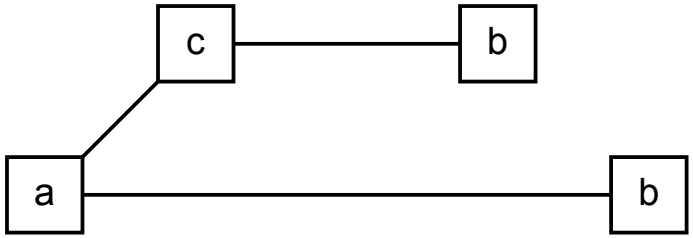
problemclass #1
27 generating gene trees

Artificial Data Analysis

Question 2 : Can one infer information about the skeleton?

Consider $\widehat{R}$ with leaf distance d as additional information.

d(x,y) =
length of the shortest path
between x and y

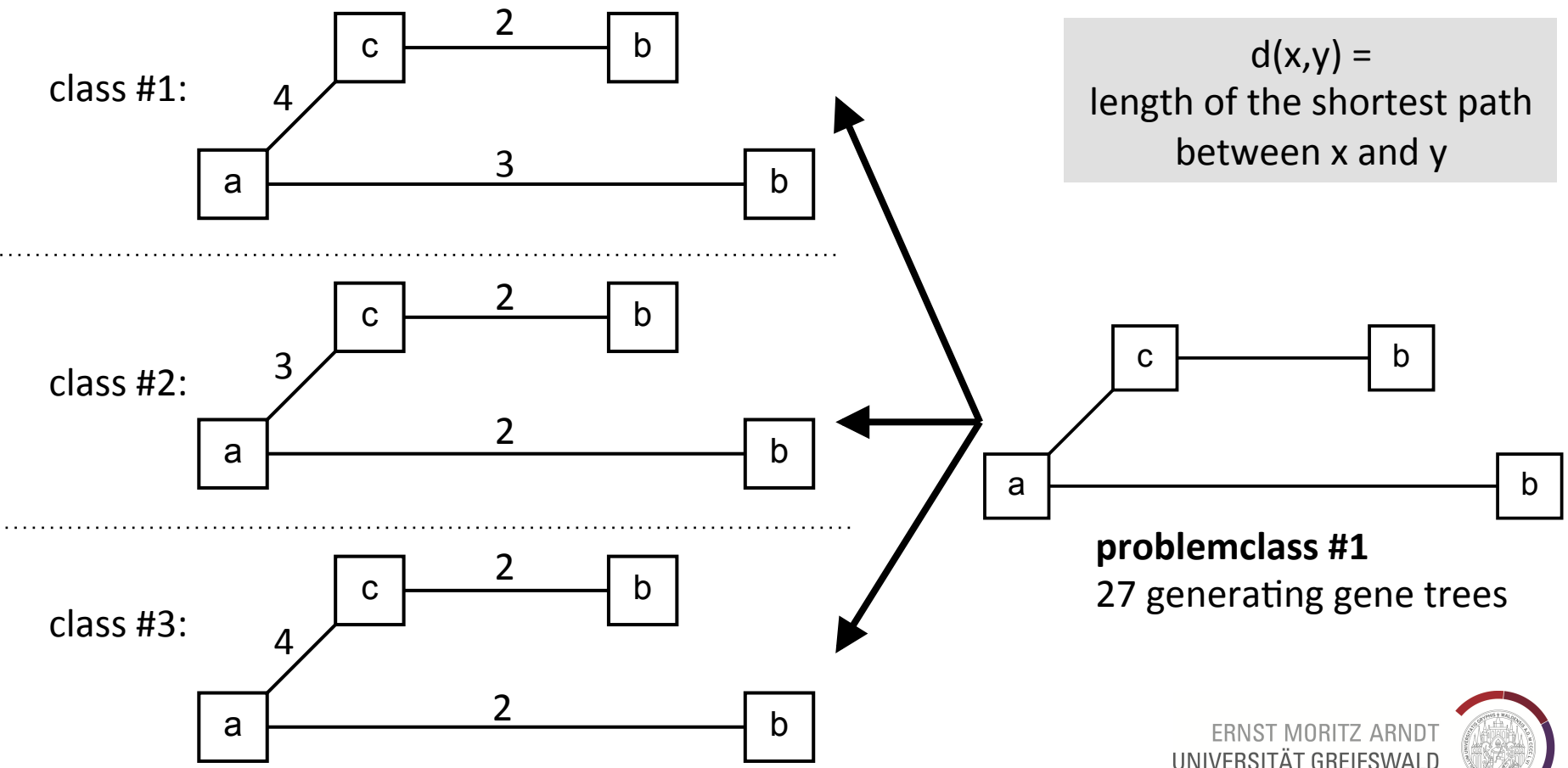


problemclass #1
27 generating gene trees

Artificial Data Analysis

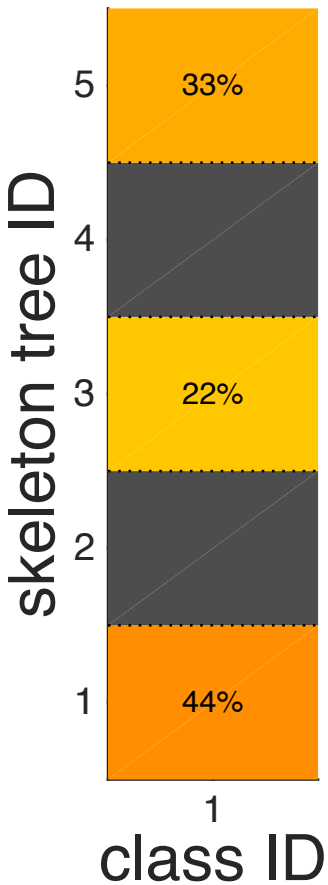
Question 2 : Can one infer information about the skeleton?

Consider $\widehat{R}$ with leaf distance d as additional information.

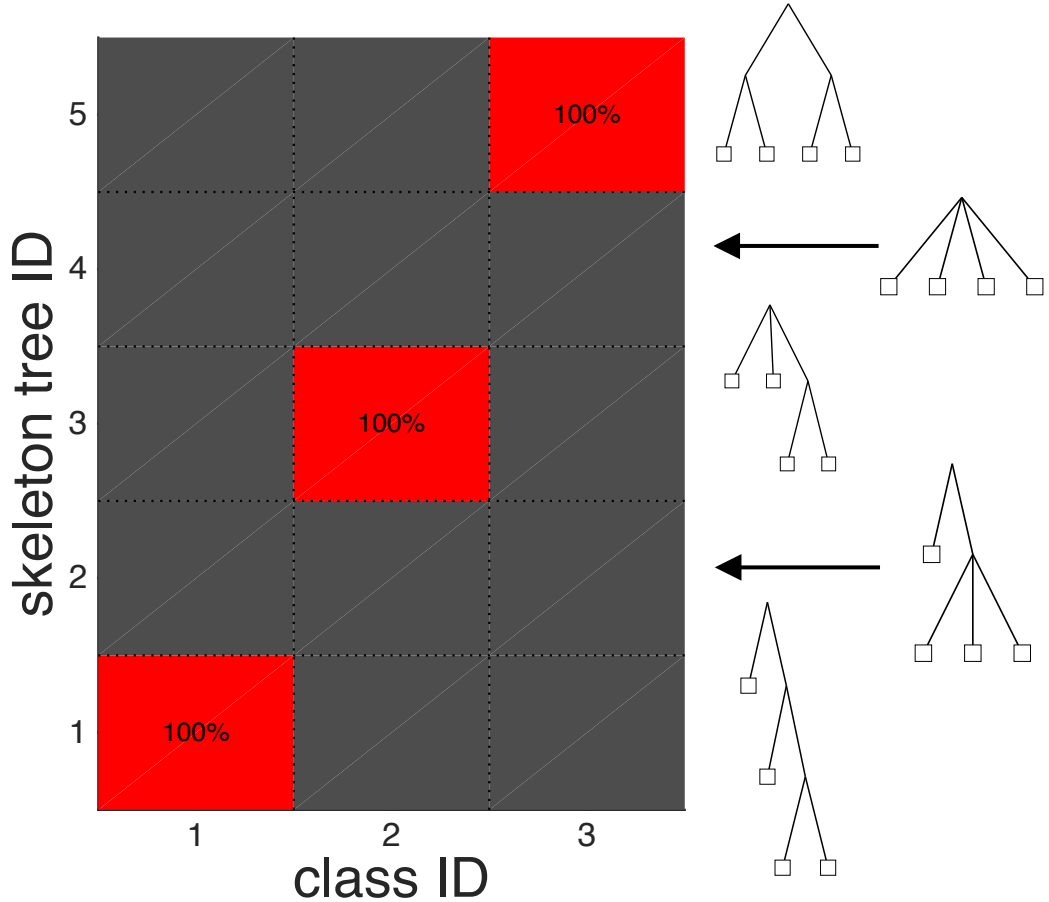


Artificial Data Analysis

Question 2 : Can one infer information about the skeleton?

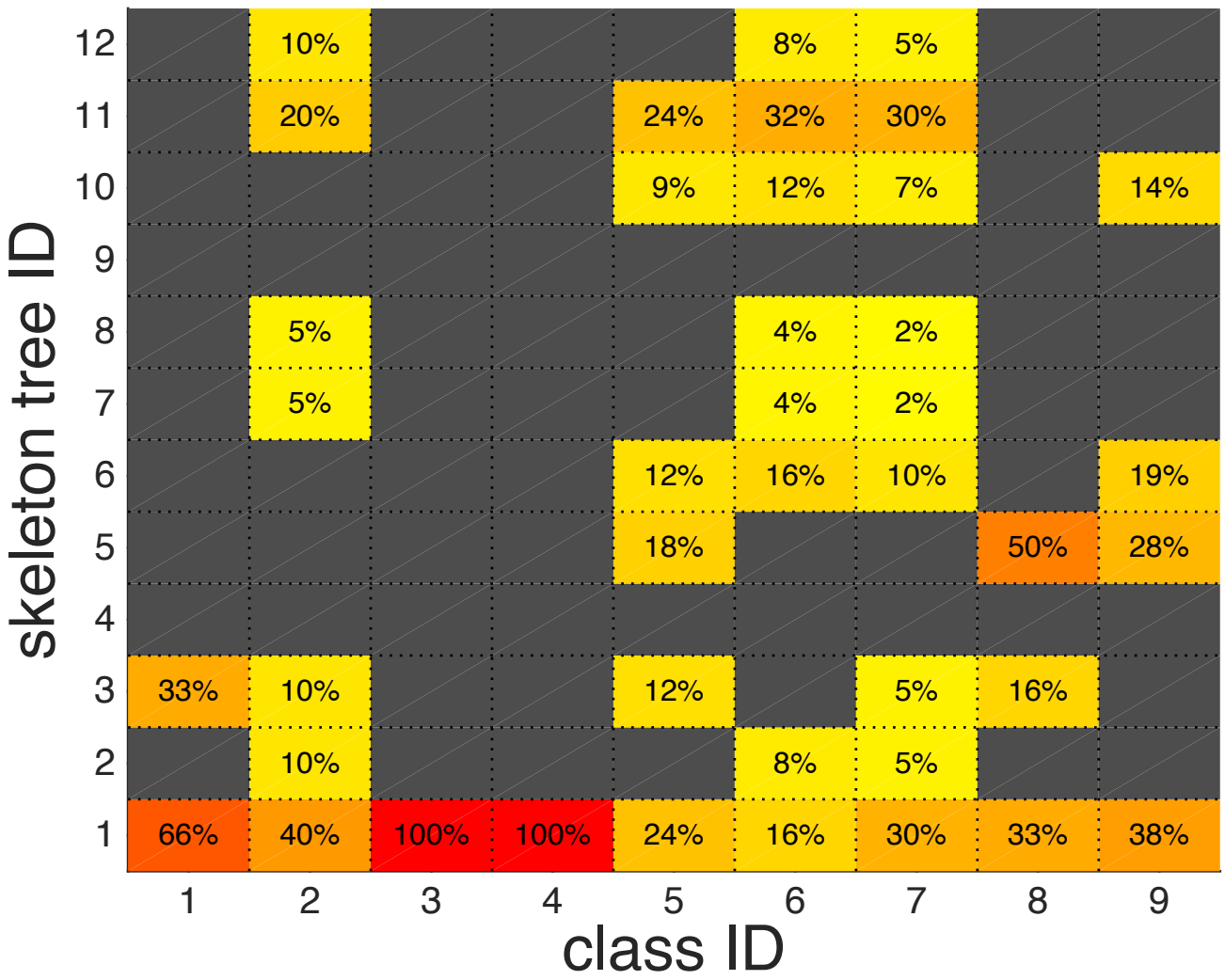


with leaf distance d

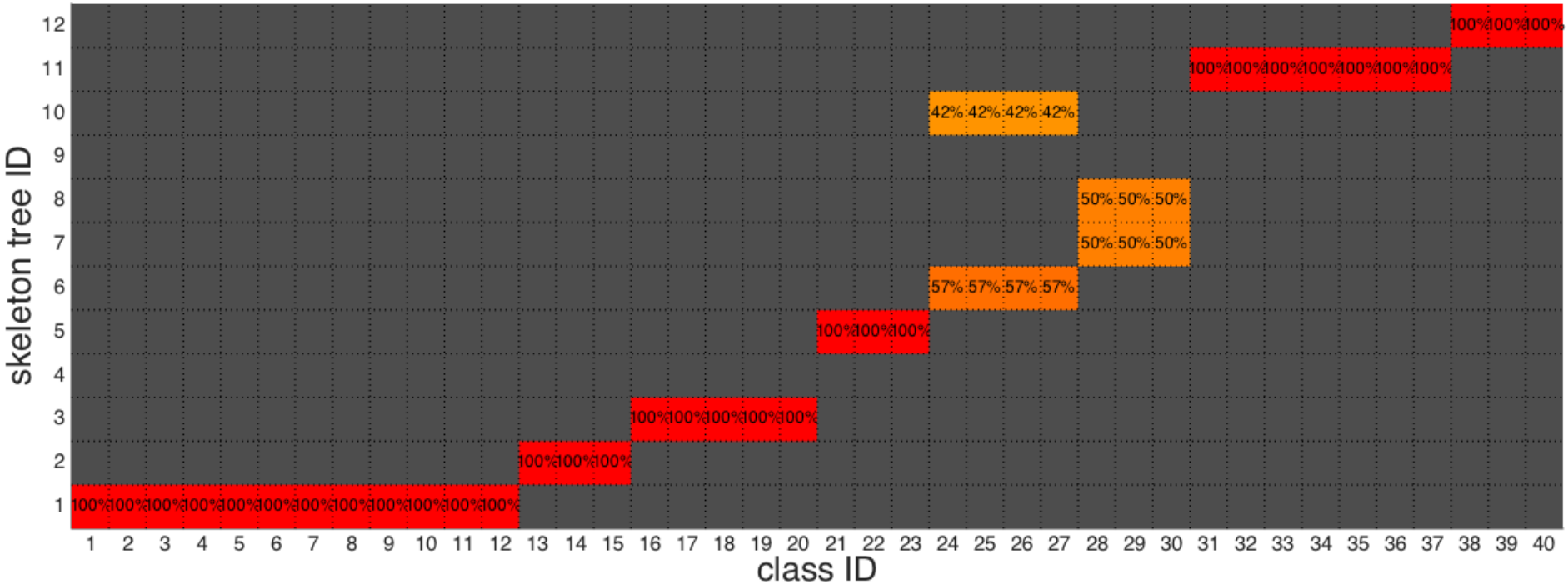


Question 2 : Can one infer information about the skeleton?

n=5

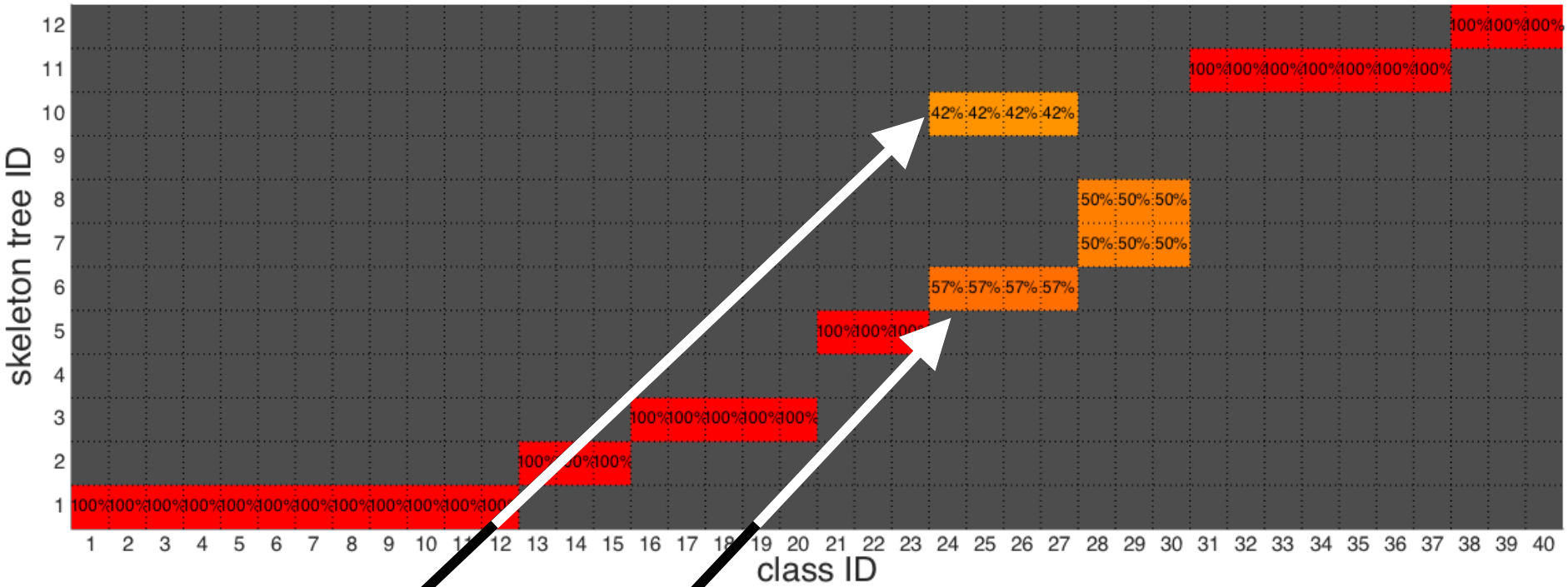


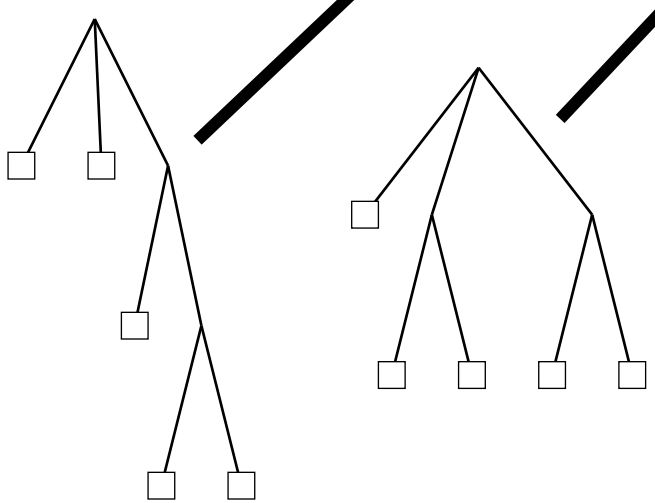
Question 2 : Can one infer information about the skeleton?



n=5 with leaf-distant d

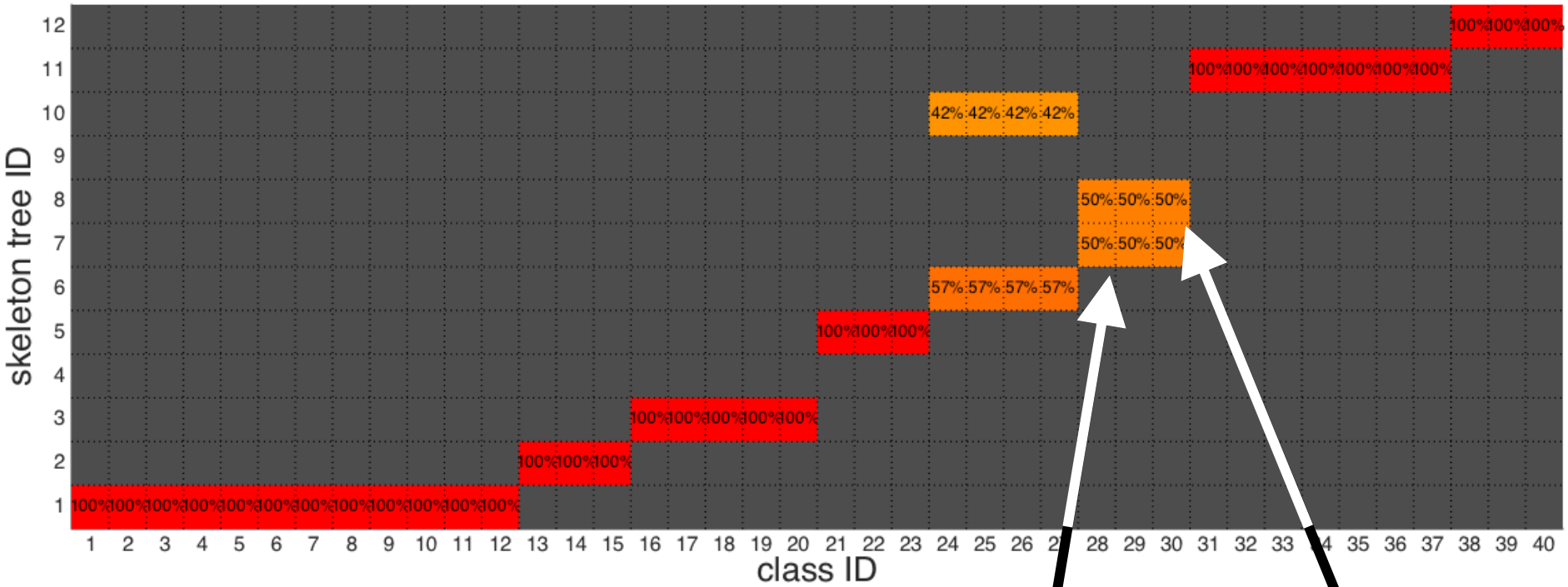
Question 2 : Can one infer information about the skeleton?

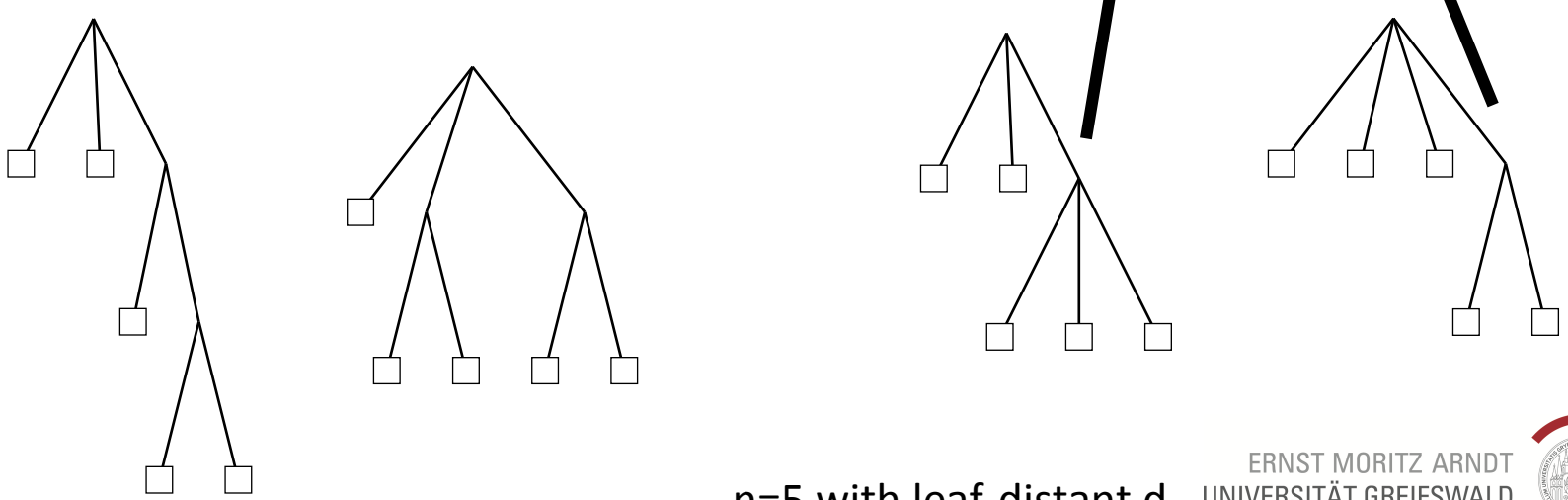




n=5 with leaf-distant d

Question 2 : Can one infer information about the skeleton?





n=5 with leaf-distant d

Summary and Outlook

Summary:

- Generated all possible gene trees for up to 7 leafs
- No information about right or wrong placed edges in the problemclasses
- The leaf distance d gives us the unrooted skeleton tree

For the future:

- Find a more efficent way to generate the leaf-labeling
- Define and investigate more questions

Summary and Outlook

Summary:

- Generated all possible gene trees for up to 7 leafs
- No information about right or wrong placed edges in the problemclasses
- The leaf distance d gives us the unrooted skeleton tree

For the future:

- Find a more efficent way to generate the leaf-labeling
- Define and investigate more questions

THANK YOU!

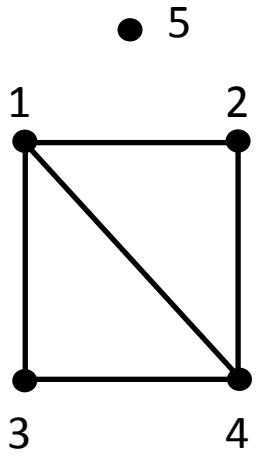
Summary and Outlook

THANK YOU!

P4-free Graphs / Cographs

A graph is called **P4-free / cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

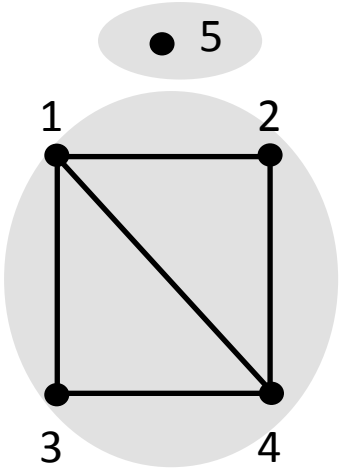
Every cograph is uniquely associated with a **cotree**
= construction instruction to create the cograph

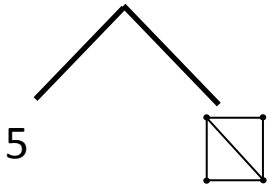


P4-free Graphs / Cographs

A graph is called **P4-free / cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

Every cograph is uniquely associated with a **cotree**
= construction instruction to create the cograph

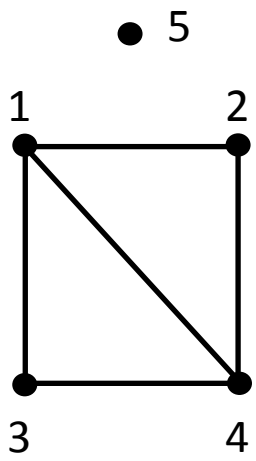


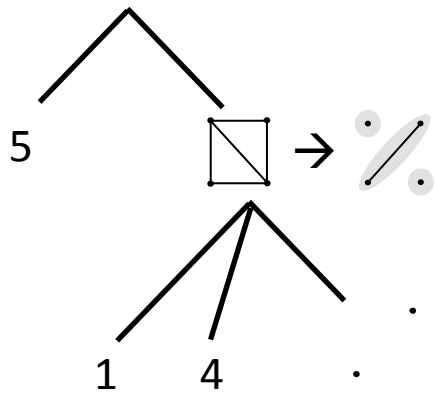


P4-free Graphs / Cographs

A graph is called **P4-free / cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

Every cograph is uniquely associated with a **cotree**
= construction instruction to create the cograph

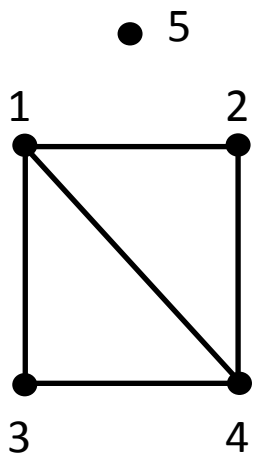


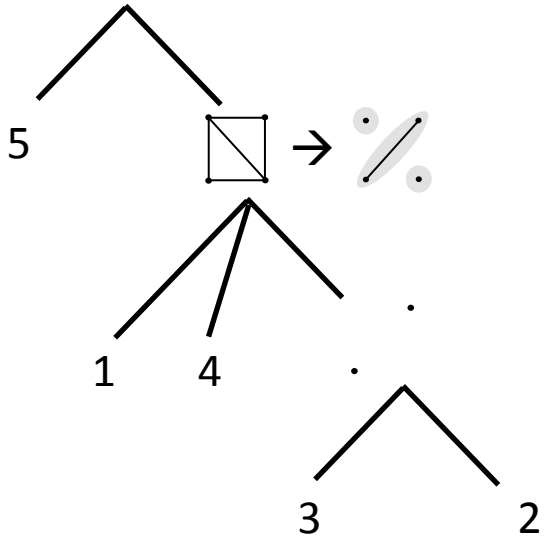


P4-free Graphs / Cographs

A graph is called **P4-free / cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

Every cograph is uniquely associated with a **cotree**
= construction instruction to create the cograph



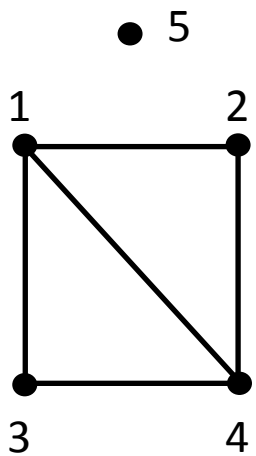


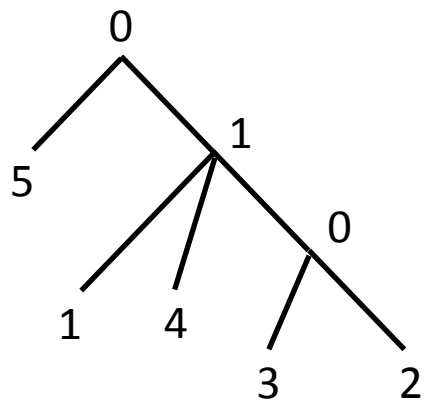
Introduction

P4-free Graphs / Cographs

A graph is called **P4-free / cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

Every cograph is uniquely associated with a **cotree**
= construction instruction to create the cograph

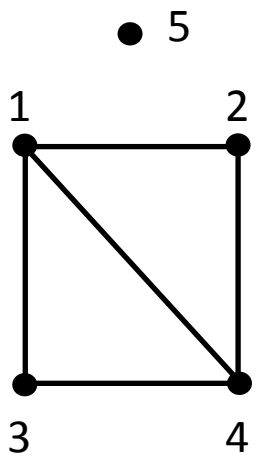


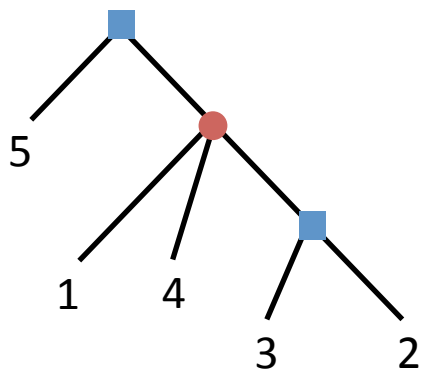


P4-free Graphs / Cographs

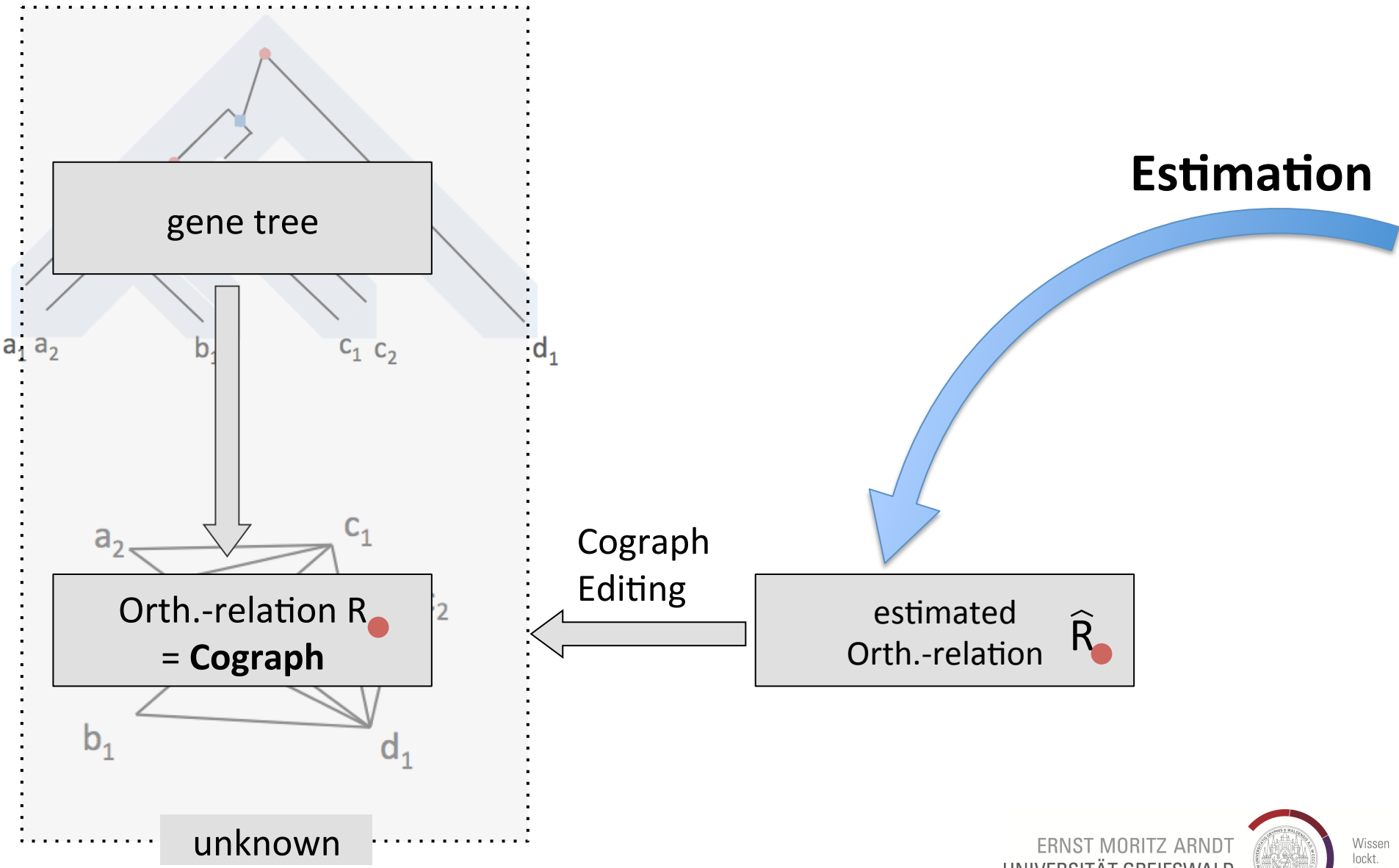
A graph is called **P4-free / cograph** if and only if
there exists no <u>induced</u> subgraph on 4 nodes that is a P4 ()

Every cograph is uniquely associated with a **cotree**
= construction instruction to create the cograph

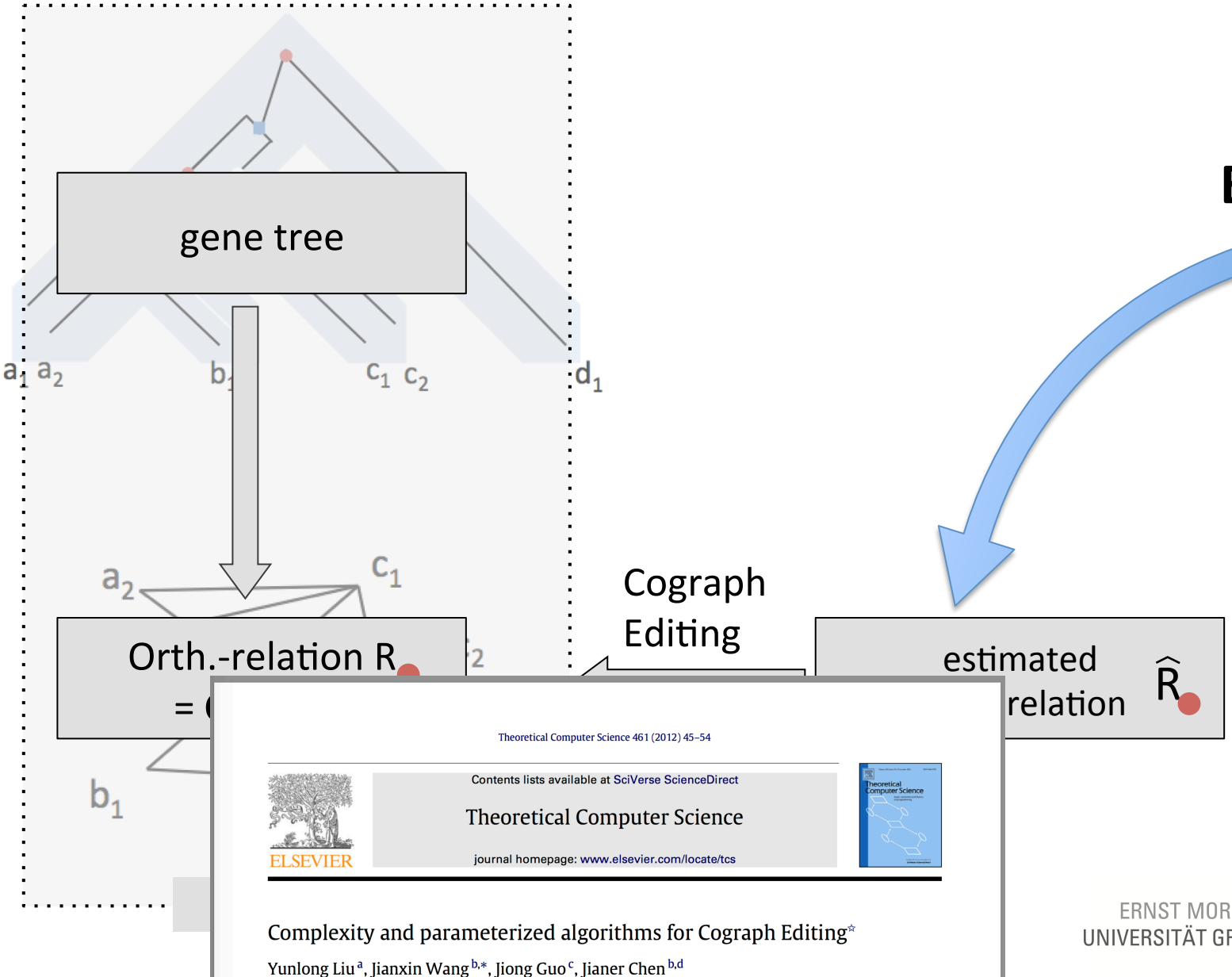




Introduction



Introduction



Theoretical Computer Science 461 (2012) 45–54

Contents lists available at SciVerse ScienceDirect

Theoretical Computer Science

journal homepage: www.elsevier.com/locate/tcs

Complexity and parameterized algorithms for Cograph Editing☆

Yunlong Liu[a], Jianxin Wang[b,*], Jiong Guo[c], Jianer Chen[b,d]

Graph-based orthology inference (e.g. proteinOrtho)

Input : sequence data

Local alignment search
(e.g. BLAST)

Sequence similarity **s(x,y)**

Then two genes x (from species A) and y (from species B) are **estimated orthologs** if:

i. A ≠ B
ii. The sequence similarity **s(x,y)** is greater than the one between
    x to all other genes of B and
    y to all other genes of A

Graph-based orthology inference (e.g. proteinOrtho)

x (from specie A) and y (from specie B) are **estimated orthologs** if:

i. $A \neq B$
ii. The sequence similarity **s(x,y)** is greater than the one between
    x to all other genes of B and
    y to all other genes of A

Simplification: leaf distance d(x,y) of the (unknown) gene tree:

d(x,y) = length of the shortest path between x and y

Graph-based orthology inference (e.g. proteinOrtho)

x (from specie A) and y (from specie B) are **estimated orthologs** if:

i. A ≠ B
ii. The sequence similarity **s(x,y)** is greater than the one between
    x to all other genes of B and
    y to all other genes of A

Simplification: leaf distance d(x,y) of the (unknown) gene tree:

d(x,y) = length of the shortest path between x and y

$$d \sim 1/s$$

Introduction

Graph-based orthology inference (e.g. proteinOrtho)

x (from specie A) and y (from specie B) are **estimated orthologs** if:

i. A ≠ B
ii. The sequence similarity **s(x,y)** is greater than the one between
    x to all other genes of B and
    y to all other genes of A

Simplification: leaf distance d(x,y) of the (unknown) gene tree:

d(x,y) = length of the shortest path between x and y

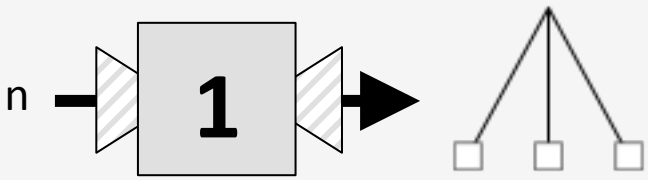
$d \sim 1/s$

→ **estimated orthologs** if:

i. A ≠ B
ii. d(x,y) is <u>smaller</u> than the distance between
    x to all other genes of B and
    y to all other genes of A

→ $\hat{R}$

Artificial Data Analysis



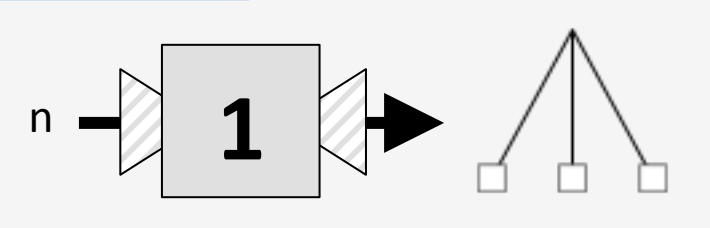
1. Generate skeleton trees

n = 3

$1 \cdot 3 \quad + \quad 2 \cdot 0 \quad = 3$

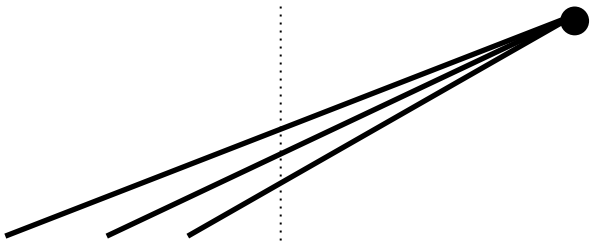
$1 \cdot 1 \quad + \quad 2 \cdot 1 \quad = 3$

Artificial Data Analysis

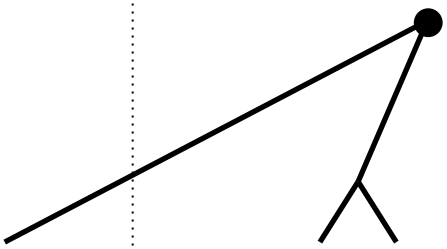


1. Generate skeleton trees

n = 3

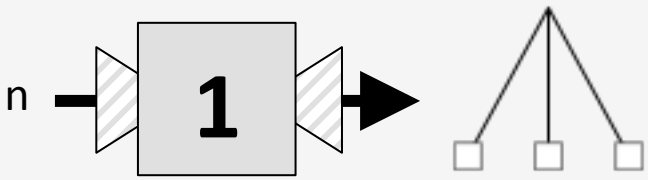


$1 \cdot 3 \quad + \quad 2 \cdot 0 \quad = 3$

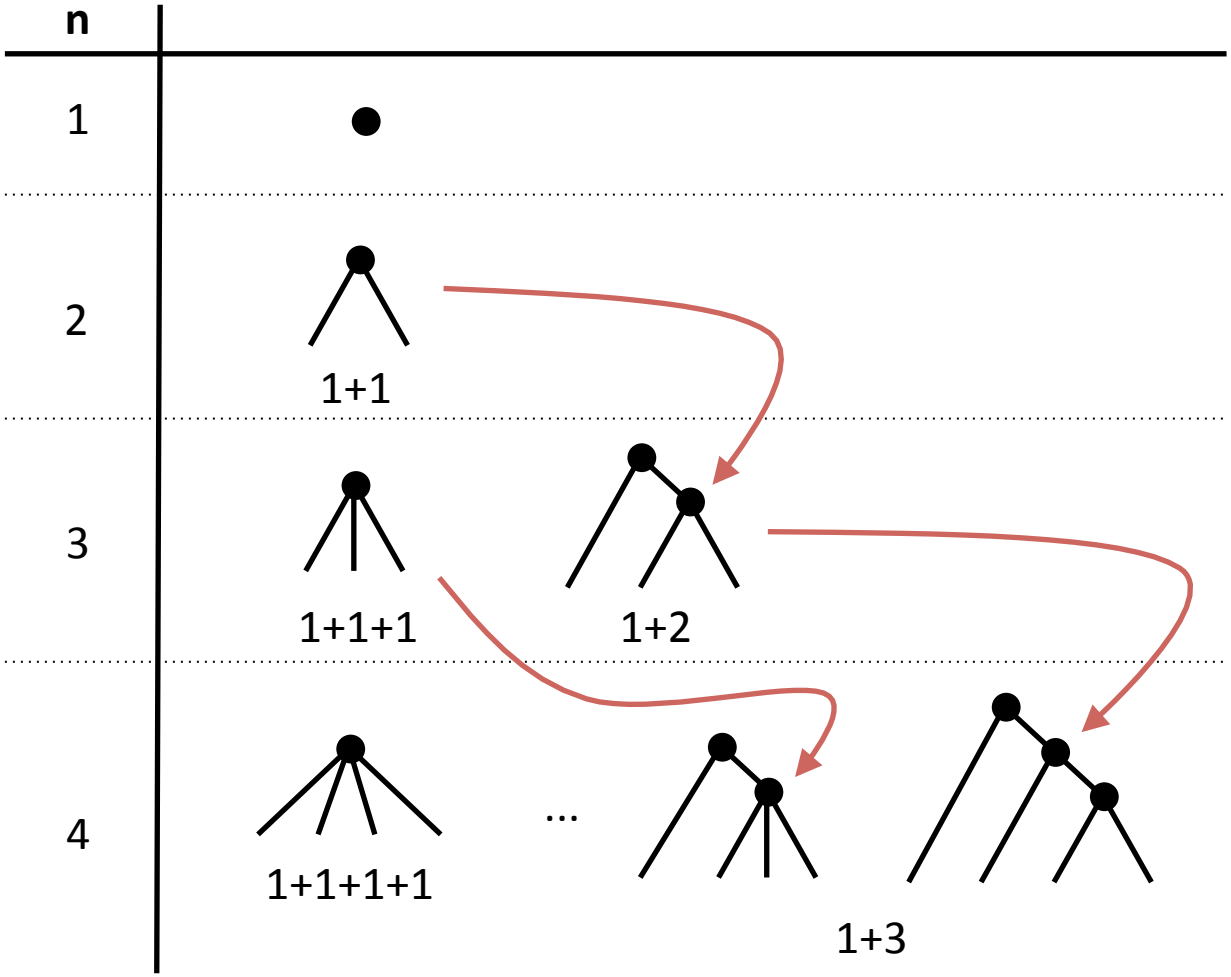


$1 \cdot 1 \quad + \quad 2 \cdot 1 \quad = 3$

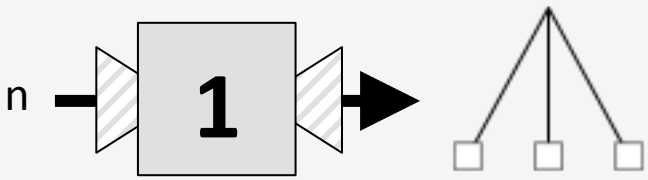
Artificial Data Analysis



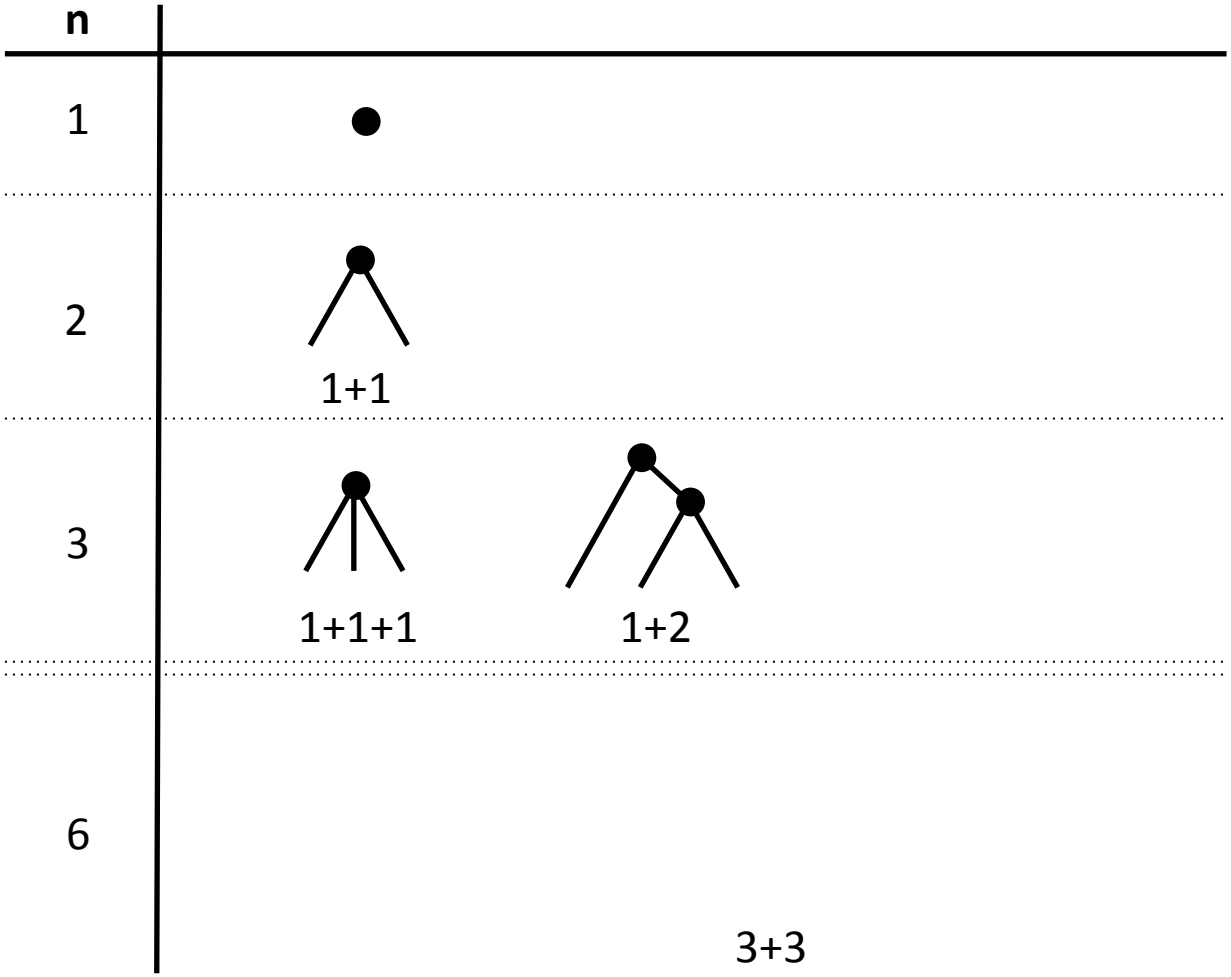
1. Generate skeleton trees



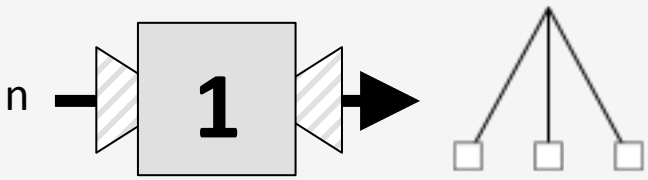
Artificial Data Analysis



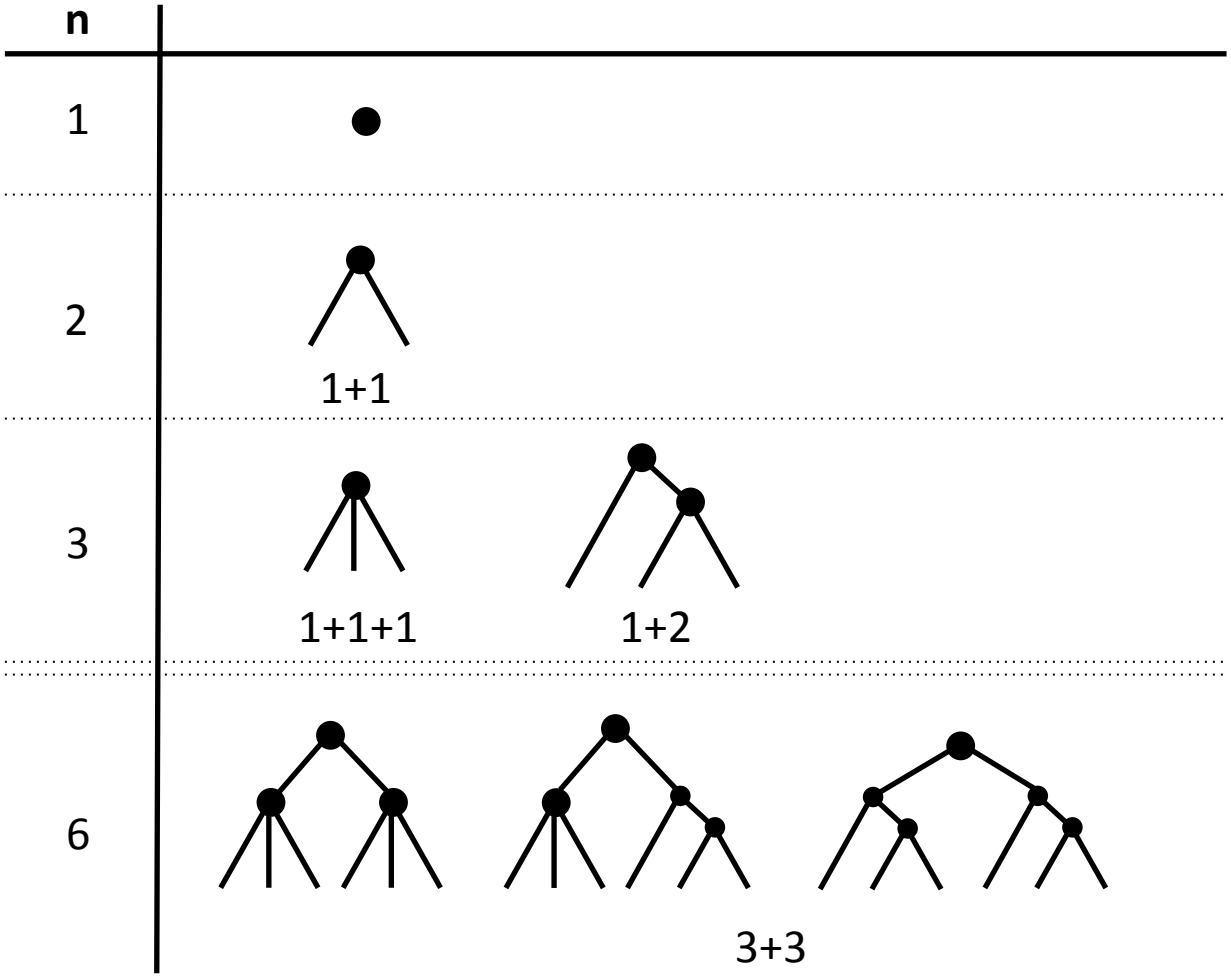
1. Generate skeleton trees



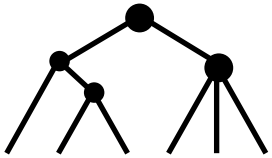
Artificial Data Analysis



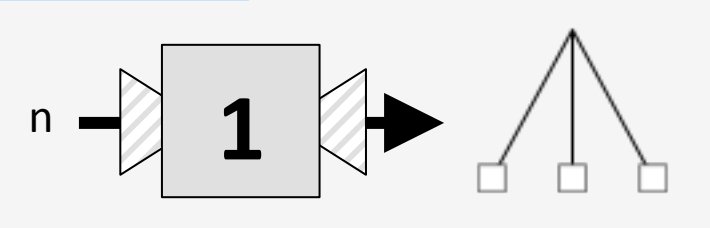
1. Generate skeleton trees



we do not want:



Artificial Data Analysis

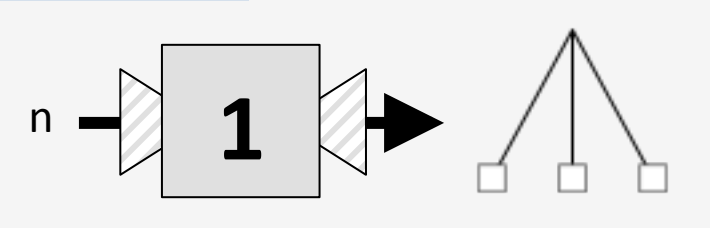


1. Generate skeleton trees

→ Choosing a_i **unordered** elements of the skeleton trees with i leafs **with** replacement

$$\binom{numST_i + (a_i - 1)}{a_i}$$

Artificial Data Analysis

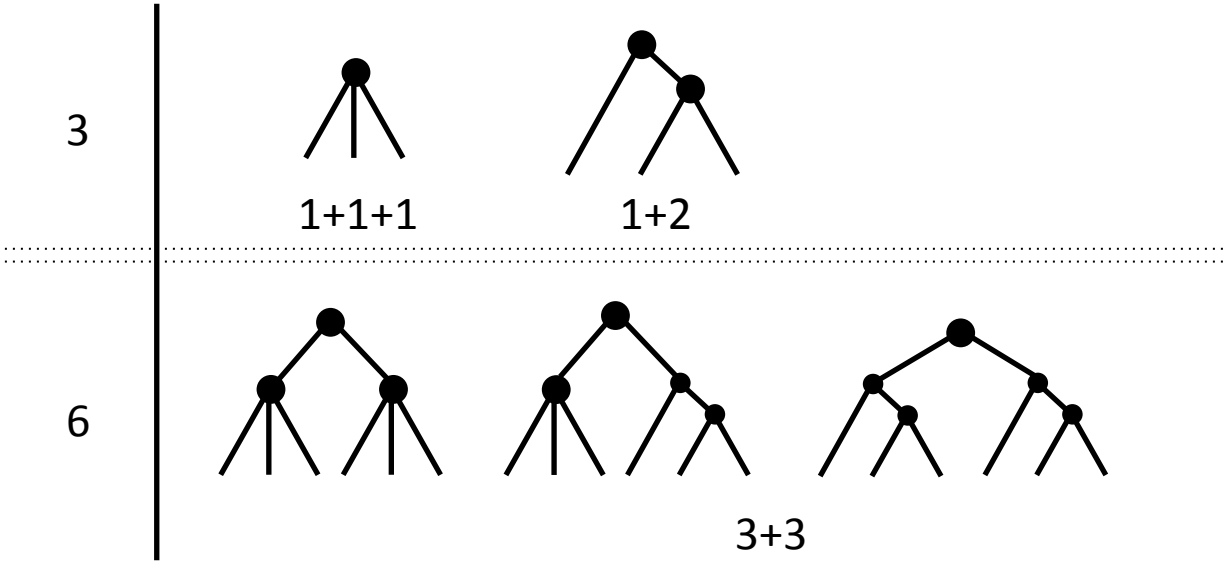


1. Generate skeleton trees

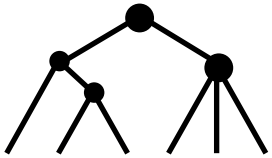
→ Choosing a_i **unordered** elements of the skeleton trees with i leafs **with** replacement

→ Choosing <u>2</u> **unordered** elements of the skeleton trees with <u>3</u> leafs **with** replacement

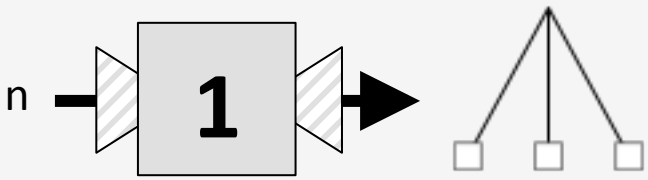
$$\binom{numST_i + (a_i - 1)}{a_i} = 3$$



we do not want:

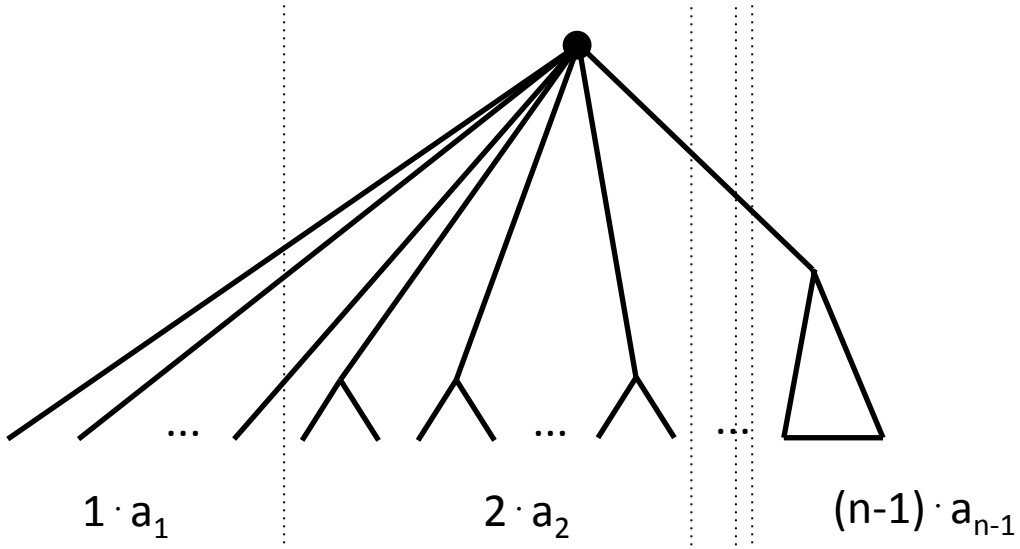


Artificial Data Analysis



1. Generate skeleton trees

$$numST_n = \sum_{(a_1,a_2,\ldots,a_{n-1})\in P^n} \prod_{i=1}^{n-1} \binom{numST_i + (a_i - 1)}{a_i}$$



Artificial Data Analysis

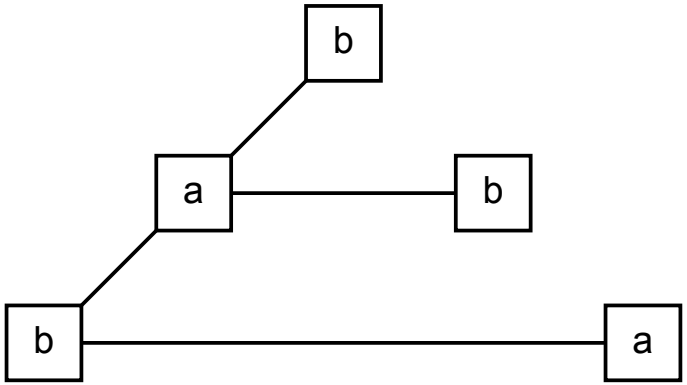
Results for n=5 leafs

Total number of gene trees : **3543**

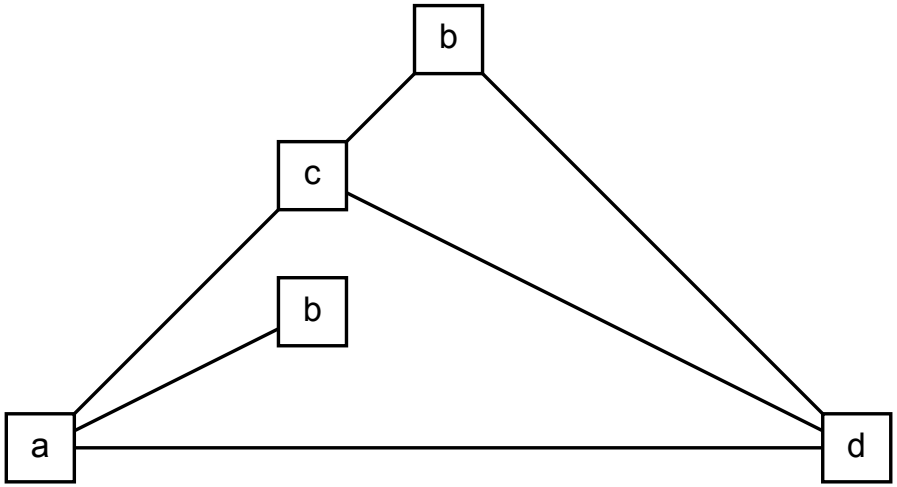
Non-cograph cases : **822**

Number of problemclasses : **9**

$\hat{R}_{\bullet}$:



problemclass #3
24 generating gene trees



problemclass #7
234 generating gene trees

Question 2 : Can one infer information about the skeleton?

Theorem:

Given the leaf distance d_1 of the skeleton T_1 and d_2 of T_2 then:

$$d_1 = d_2 \quad \Rightarrow \quad \text{Unroot}(\, T_1 \,) \sim \text{Unroot}\,(\, T_2 \,)$$

Unroot(T) : unrooted version of the tree T