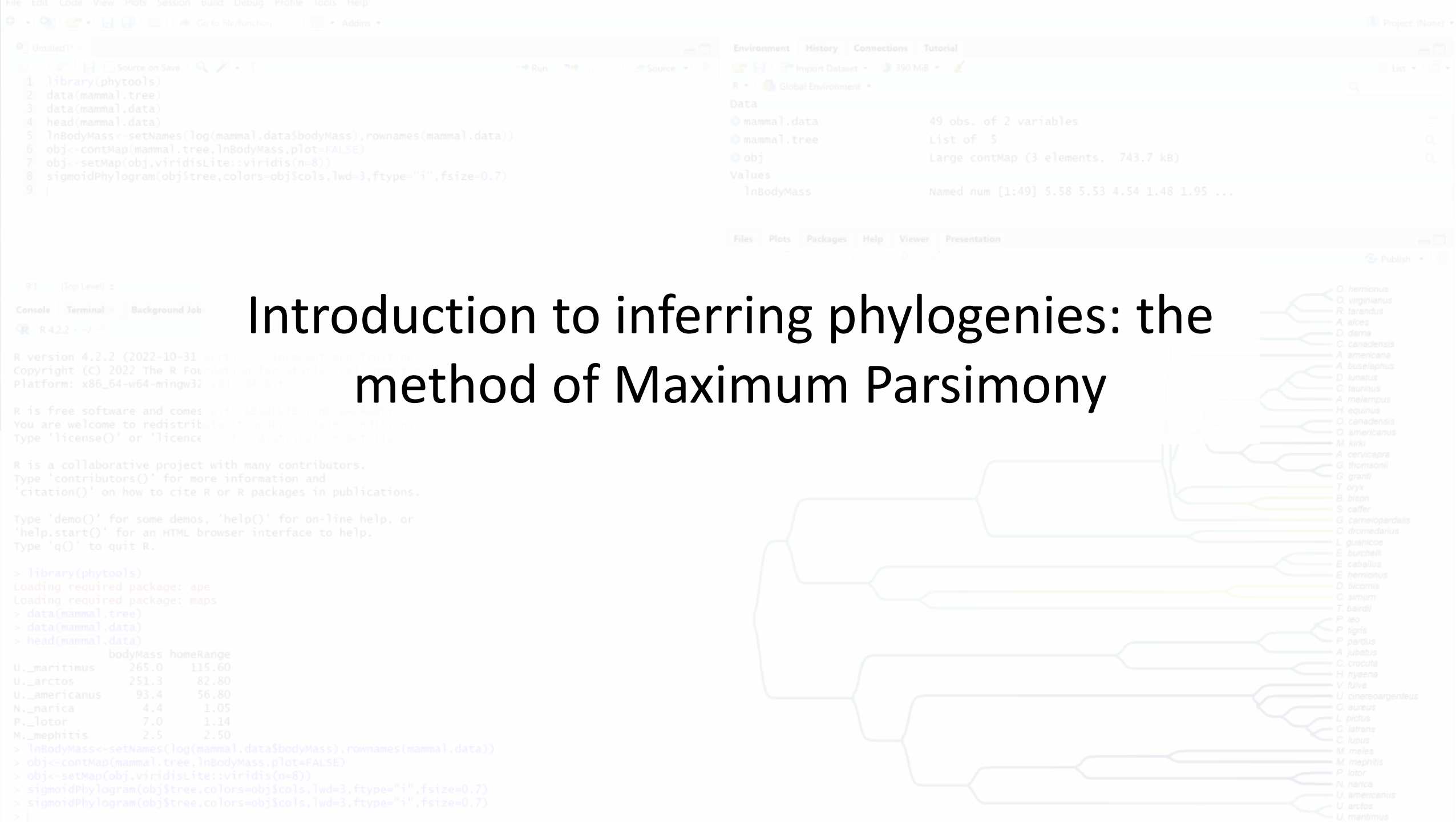
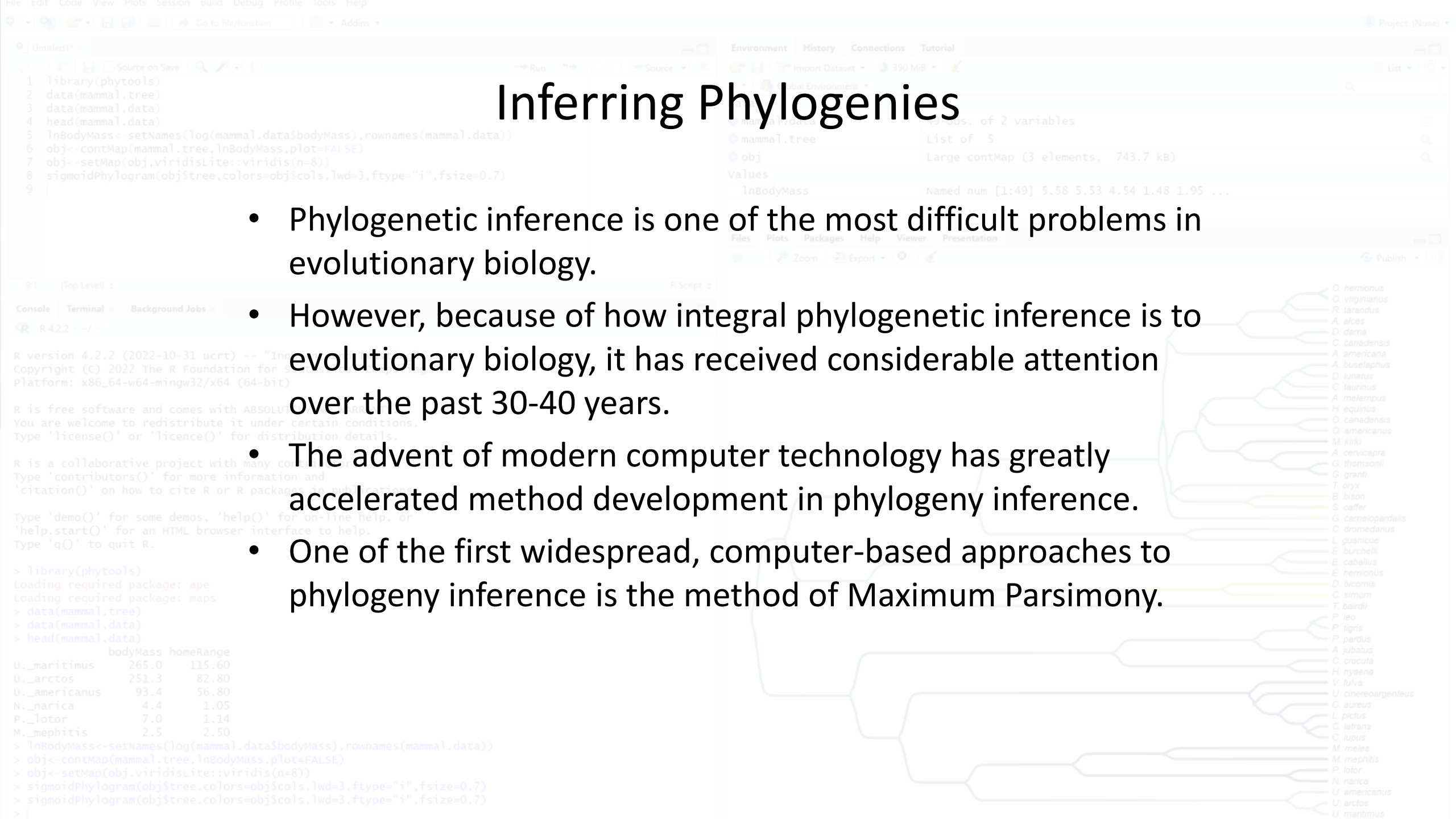


# Introduction to inferring phylogenies: the method of Maximum Parsimony



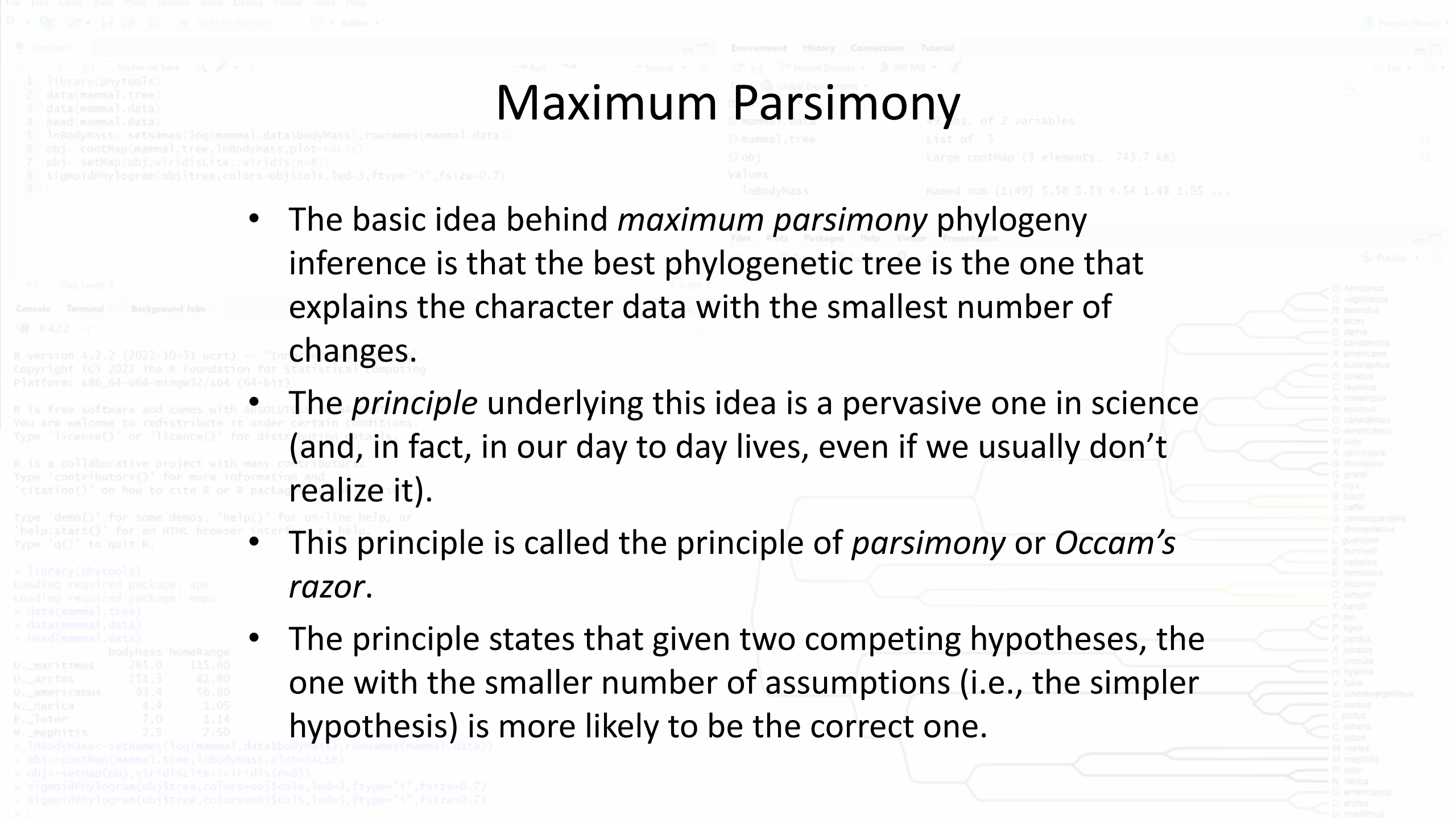
# Inferring Phylogenies

- Phylogenetic inference is one of the most difficult problems in evolutionary biology.
- However, because of how integral phylogenetic inference is to evolutionary biology, it has received considerable attention over the past 30-40 years.
- The advent of modern computer technology has greatly accelerated method development in phylogeny inference.
- One of the first widespread, computer-based approaches to phylogeny inference is the method of Maximum Parsimony.



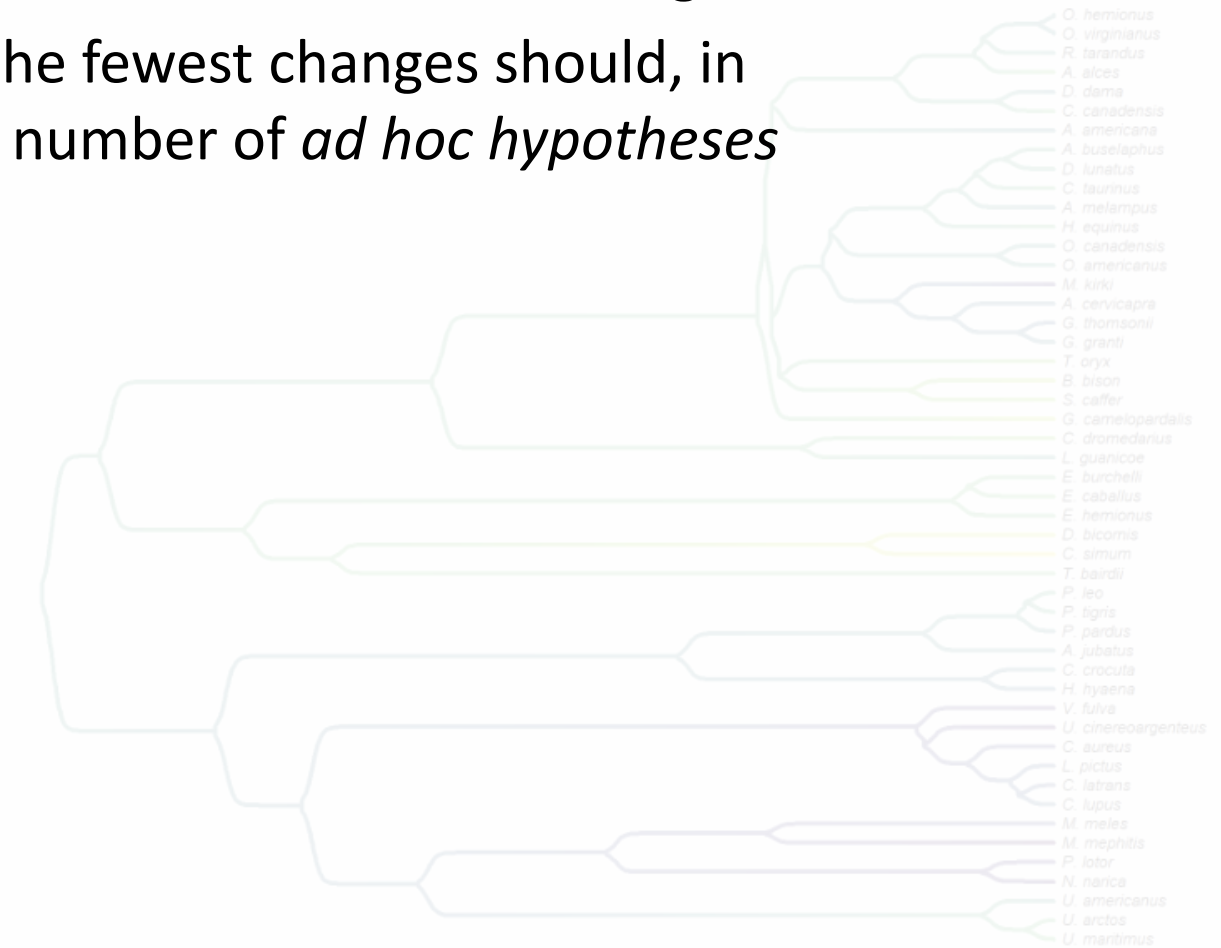
# Maximum Parsimony

- The basic idea behind *maximum parsimony* phylogeny inference is that the best phylogenetic tree is the one that explains the character data with the smallest number of changes.
- The *principle* underlying this idea is a pervasive one in science (and, in fact, in our day to day lives, even if we usually don't realize it).
- This principle is called the principle of *parsimony* or *Occam's razor*.
- The principle states that given two competing hypotheses, the one with the smaller number of assumptions (i.e., the simpler hypothesis) is more likely to be the correct one.

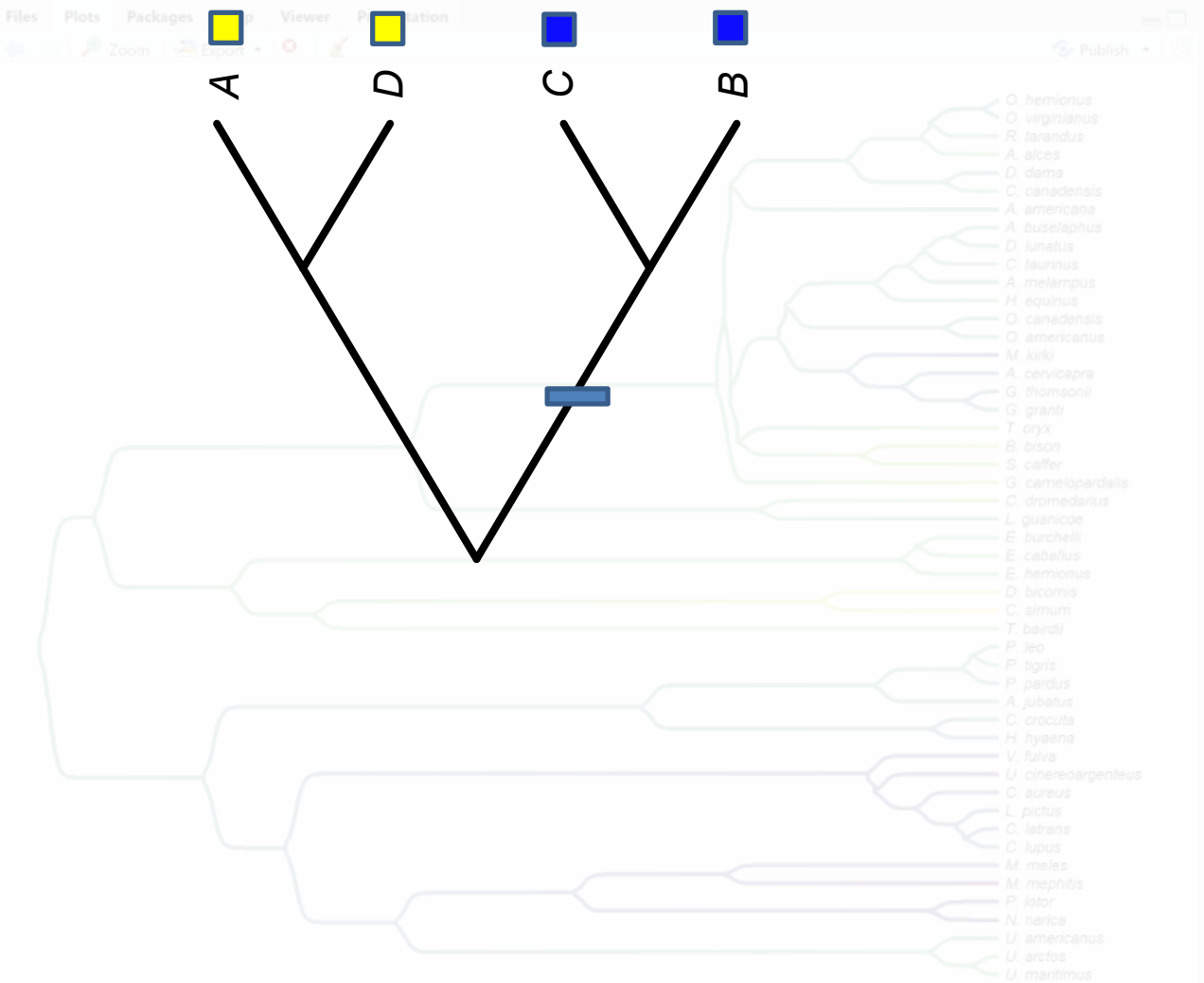
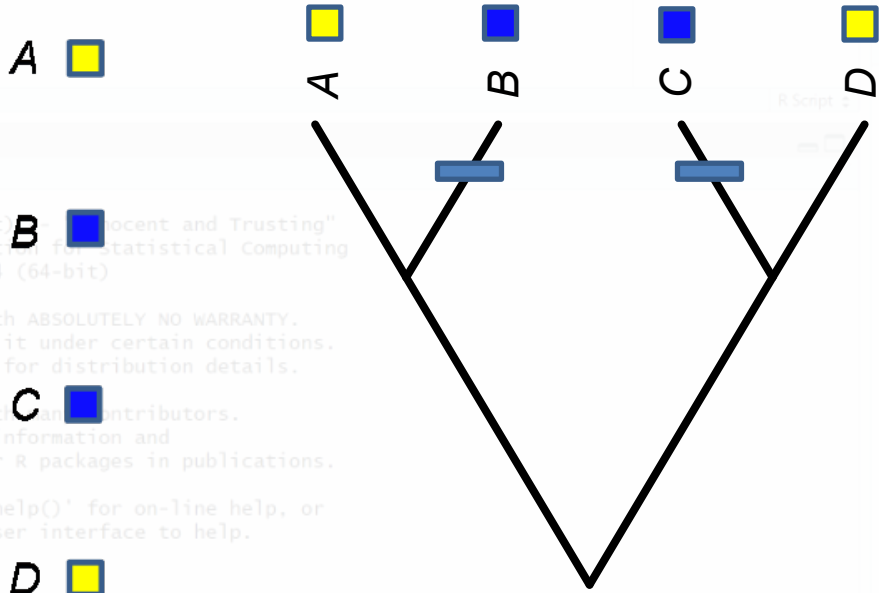


# Maximum Parsimony

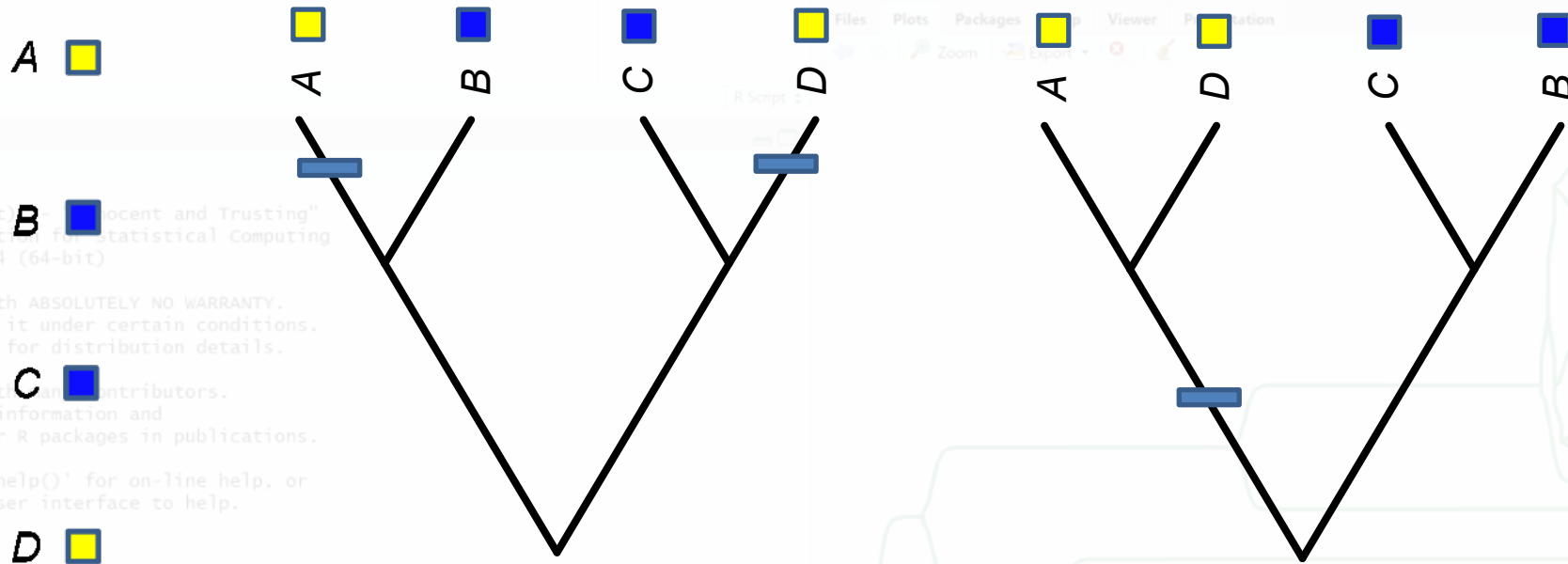
- Maximum parsimony phylogeny inference involves choosing the phylogenetic tree with the smallest number of changes.
- This is because the tree with the fewest changes should, in principle, involve the smallest number of *ad hoc hypotheses of homoplasy*.



# Maximum Parsimony



# Maximum Parsimony



- The tree on the right has the smaller number of changes and thus the smaller number of ad hoc hypotheses of homoplasy.



# Terminology

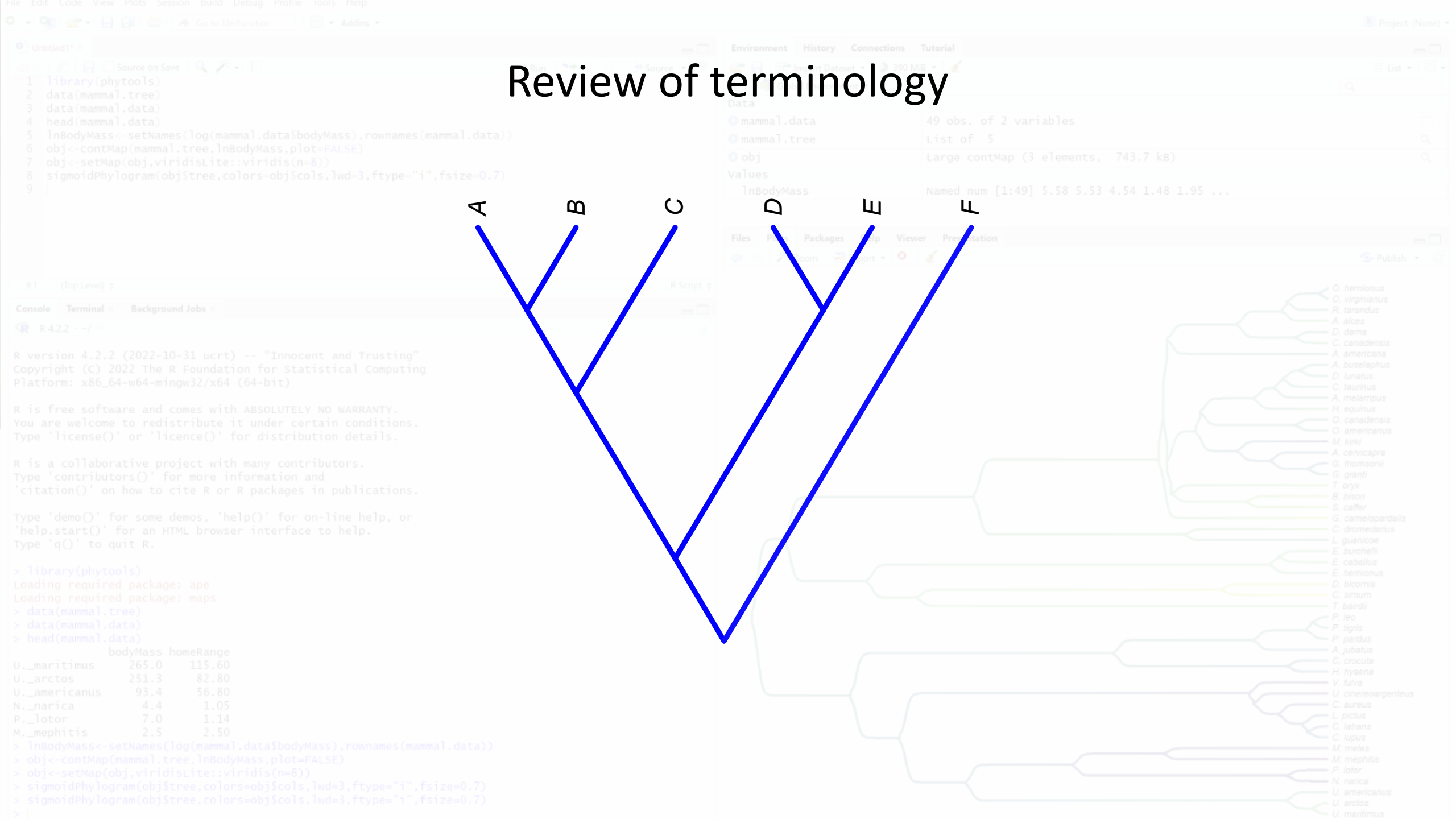
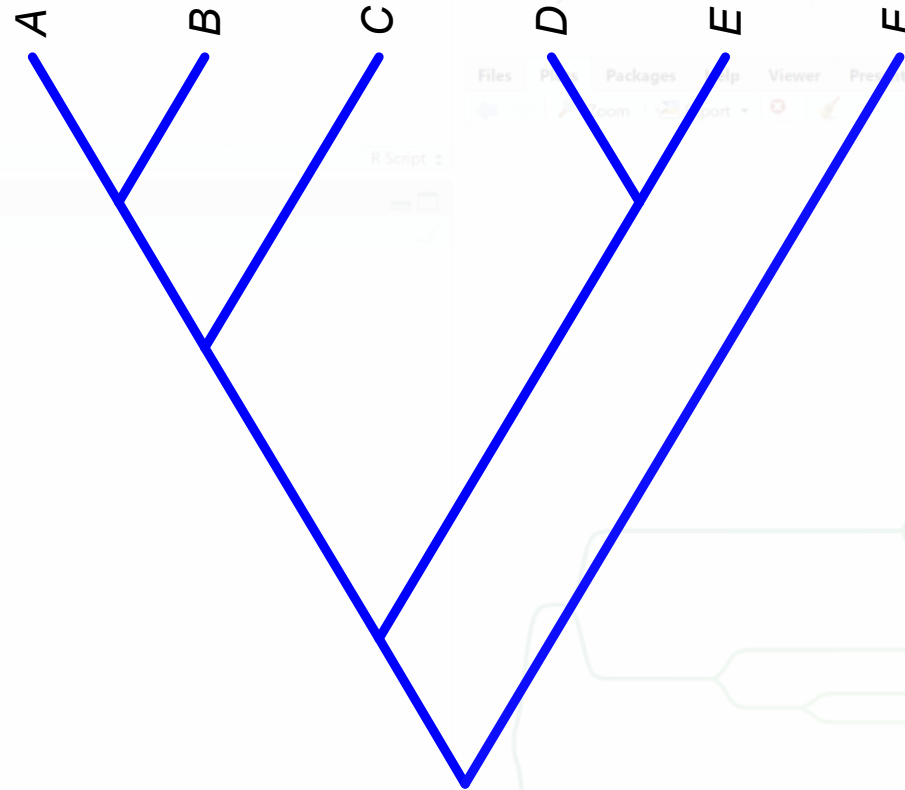
- Some additional terminology is relevant to phylogeny inference by Maximum Parsimony.
- *Plesiomorphy* refers to an ancestral character state.
- *Apomorphy* refers to a derived character state.
- *Symplesiomorphy* is a shared ancestral characteristic.
- *Synapomorphy* is a shared derived characteristic.
- Only synapomorphies are useful for parsimony phylogeny inference.

## Terminology

- *Homoplasy* refers to derived states that are similar, but derive from different ancestors. Homoplasies provide misleading information in parsimony phylogeny inference.

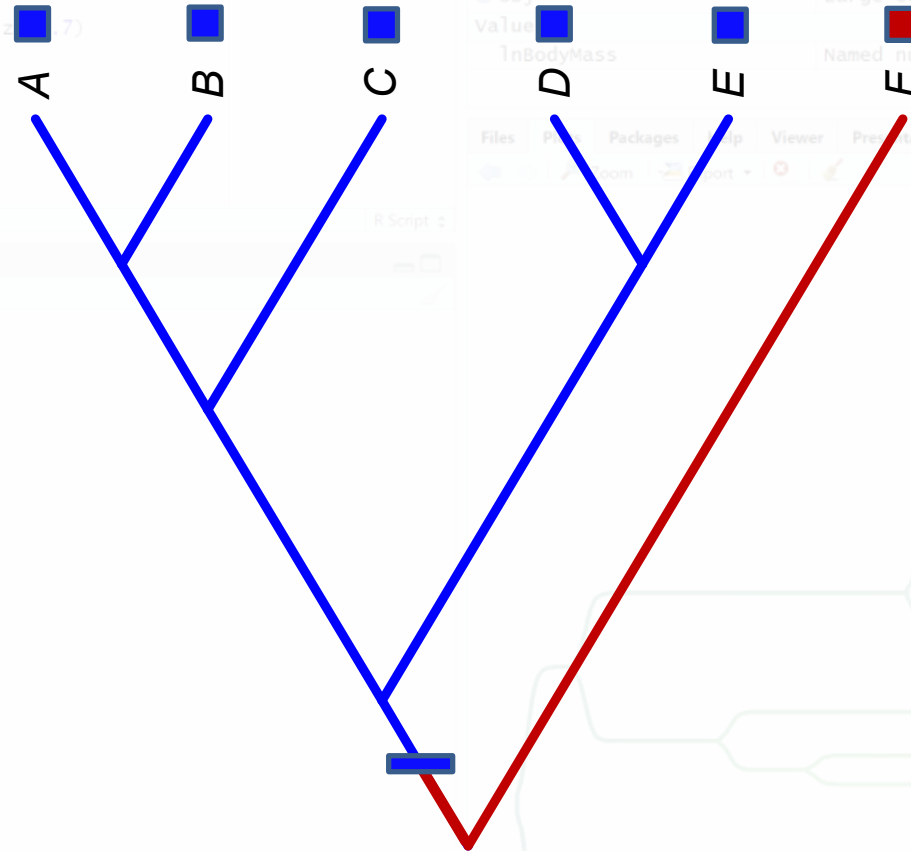
O. hemionus  
O. virginianus  
R. tarandus  
A. alces  
D. dama  
C. canadensis  
A. americana  
A. buselaphus  
D. lunatus  
C. taurinus  
A. melampus  
H. equinus  
O. canadensis  
O. americanus  
M. kirkii  
A. cervicapra  
G. thomsoni  
G. granti  
T. oryx  
B. bison  
S. caffer  
G. camelopardalis  
O. dromedarius  
L. guanicoe  
E. burchelli  
E. caballus  
E. hemionus  
D. bicornis  
C. simum  
T. bairdi  
P. leo  
P. tigris  
P. pardus  
A. jubatus  
C. crocuta  
H. hyena  
V. fulva  
U. cinereoargenteus  
C. aureus  
L. pictus  
C. latrans  
C. lupus  
M. meles  
M. mephitis  
P. lotor  
N. narica  
U. americanus  
U. arctos  
U. maritimus

# Review of terminology

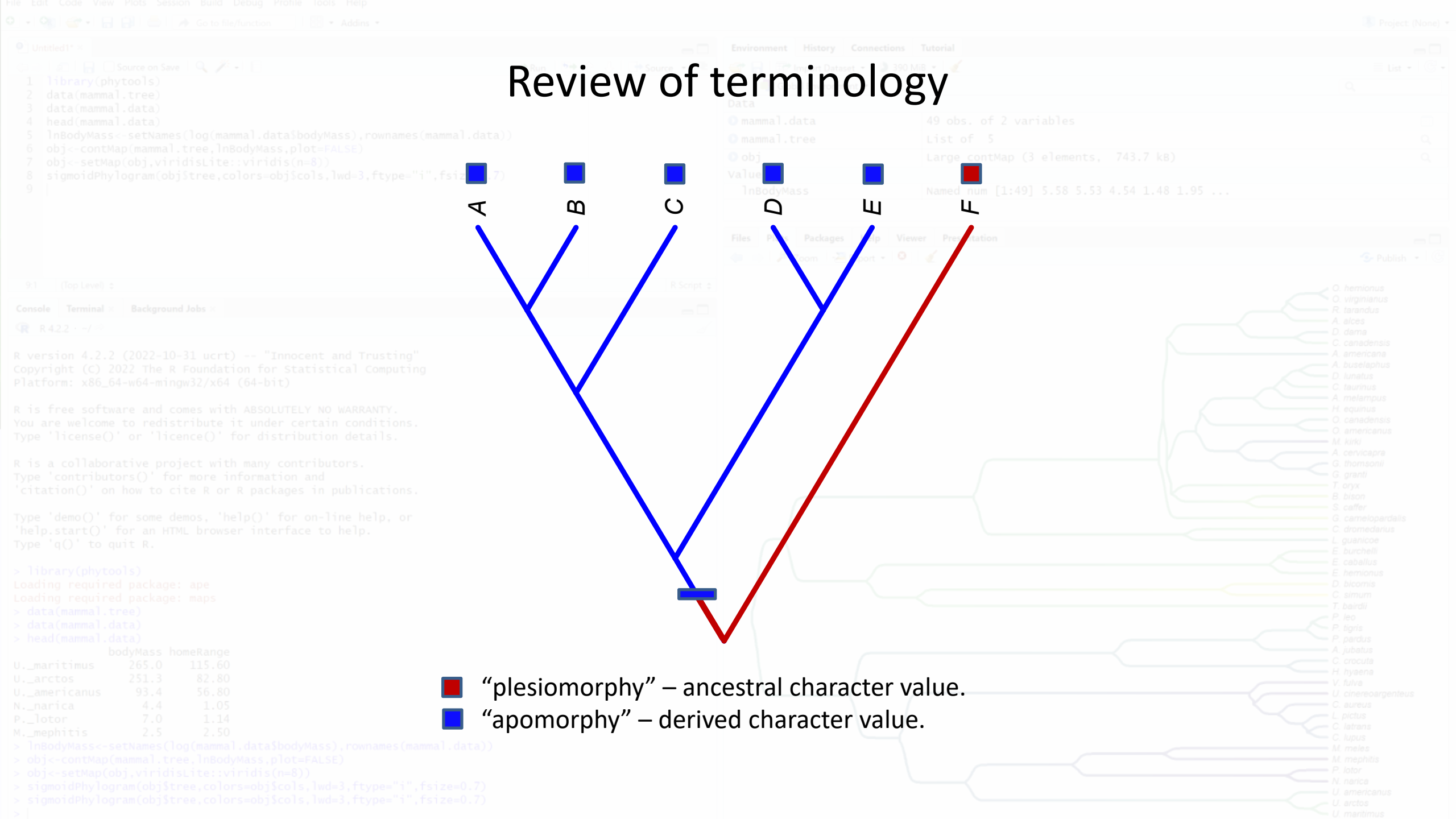




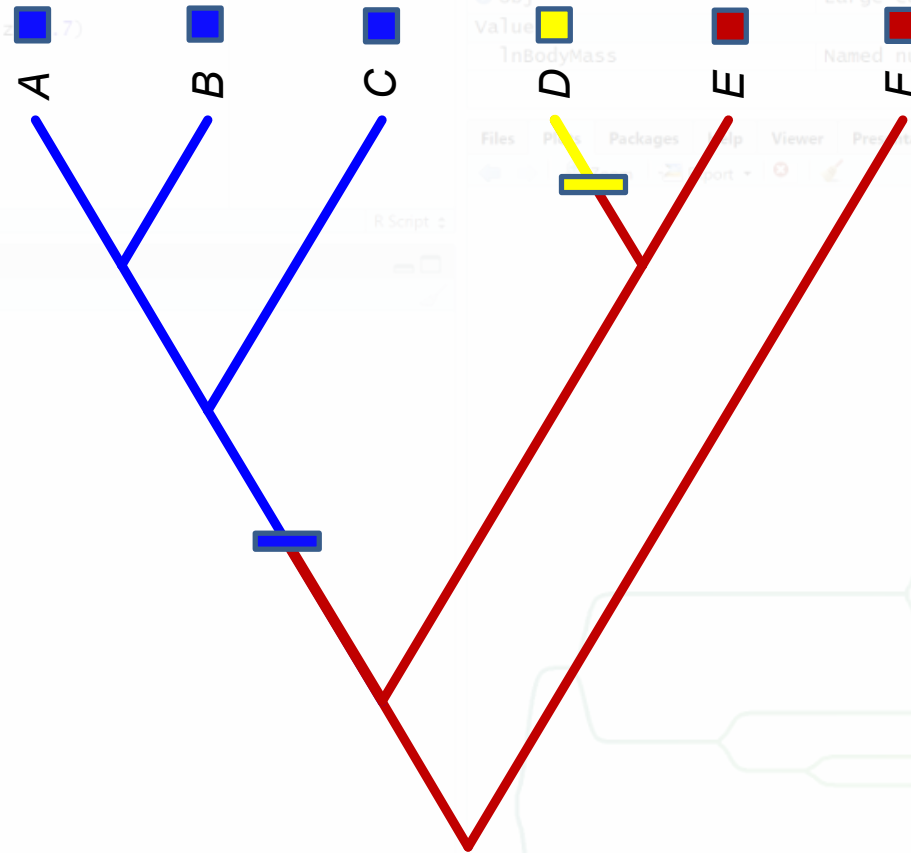
# Review of terminology



- “plesiomorphy” – ancestral character value.
- “apomorphy” – derived character value.



# Review of terminology



- “symplesiomorphy” – shared ancestral character value.
- “synapomorphy” – shared derived character value.
- “autapomorphy” – unique derived character value.

```
1 library(phytools)
2 data(mammal.tree)
3 data(mammal.data)
4 head(mammal.data)
5 lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
6 obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
7 obj<-setMap(obj,viridisLite::viridis(n=8))
8 sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftype="i",fsiz=0.7)
9
```

9.1 [Top Level] :

Console Terminal Background Jobs

R 4.2.2 - / -

R version 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"  
Copyright (C) 2022 The R Foundation for Statistical Computing  
Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

```
> library(phytools)
Loading required package: ape
Loading required package: maps
> data(mammal.tree)
> data(mammal.data)
> head(mammal.data)
```

	bodyMass	homeRange
U._maritimus	265.0	115.60
U._arctos	251.3	82.80
U._americanus	93.4	56.80
N._narica	4.4	1.05
P._lotor	7.0	1.14
M._mephitis	2.5	2.50

```
> lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
> obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
> obj<-setMap(obj,viridisLite::viridis(n=8))
> sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftype="i",fsiz=0.7)
> sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftype="i",fsiz=0.7)
>
```

Environment History Connections Tutorial

Data

mammal.data

mammal.tree

obj

Value

lnBodyMass

49 obs. of 2 variables

List of 5

Large contMap (3 elements, 743.7 kb)

Named num [1:49] 5.58 5.53 4.54 1.48 1.95 ...

Files

Plots

Packages

Help

Viewer

Presentation

Project (None)

List

Publish

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

Plot

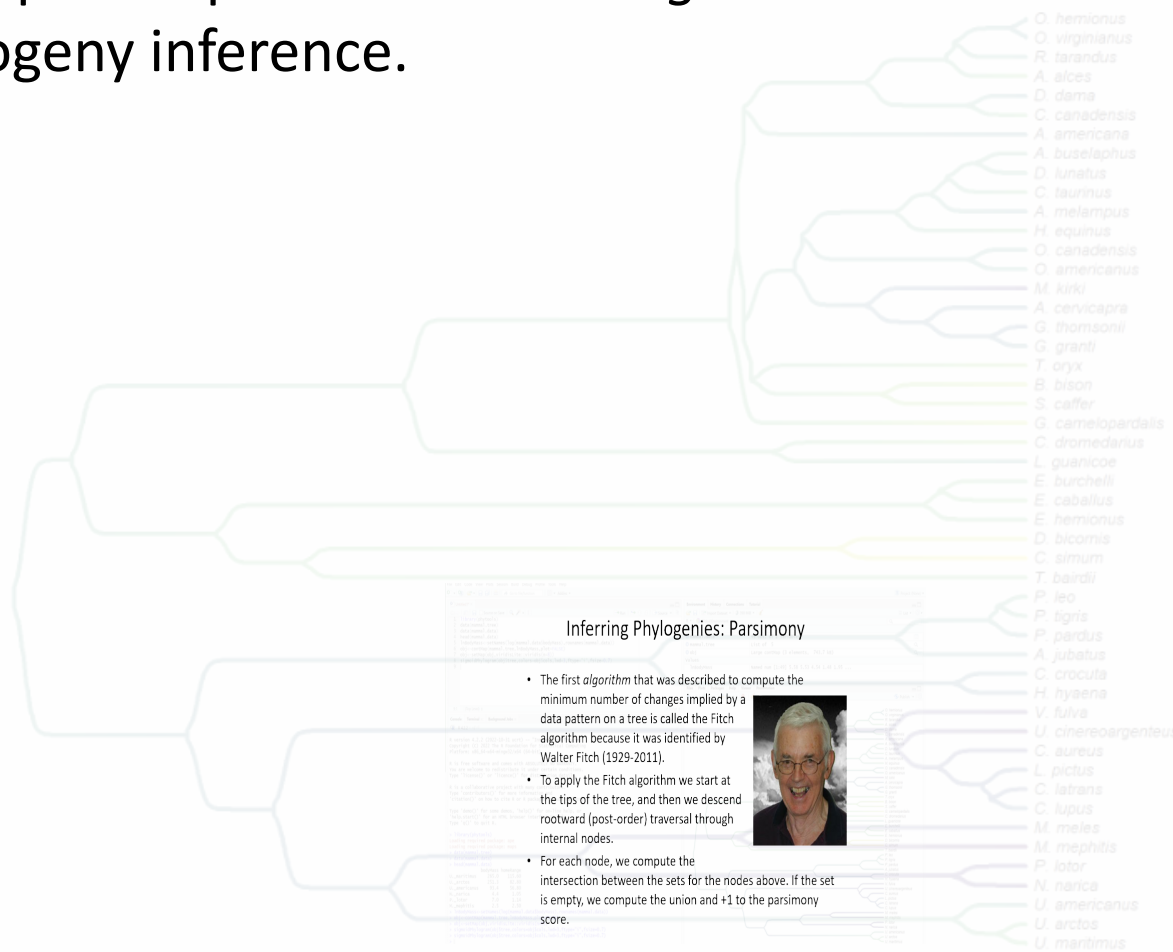
Plot

Plot

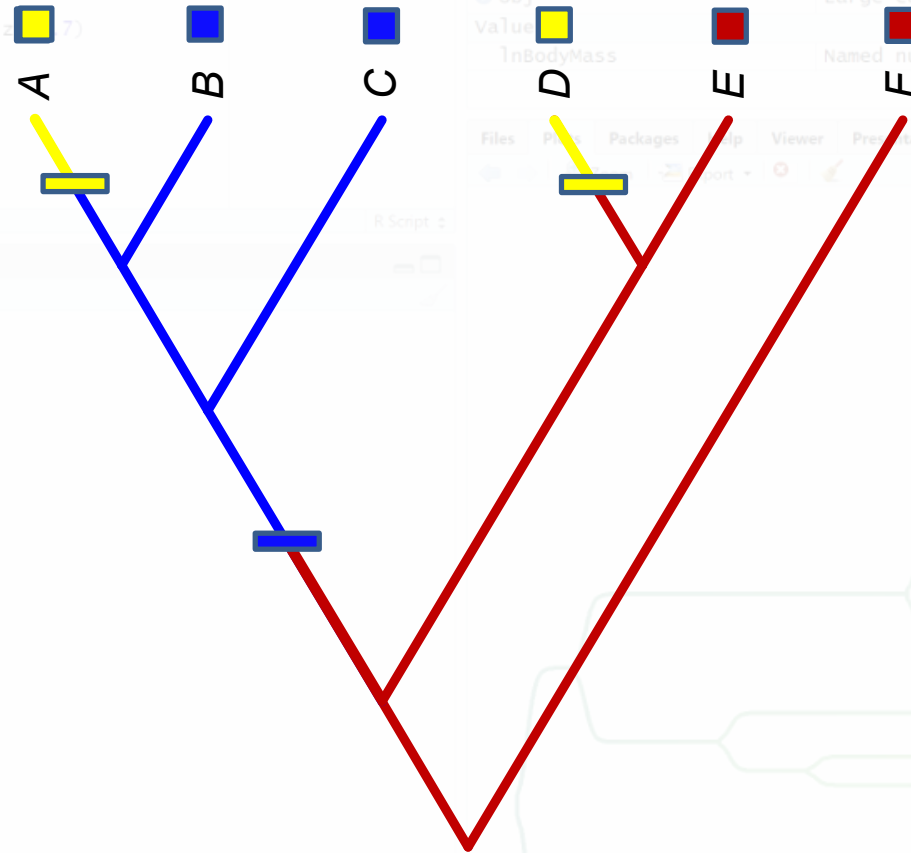
Plot

# Terminology

- *Homoplasy* refers to derived states that are similar, but derive from different ancestors. Homoplasies provide *misleading* information in parsimony phylogeny inference.



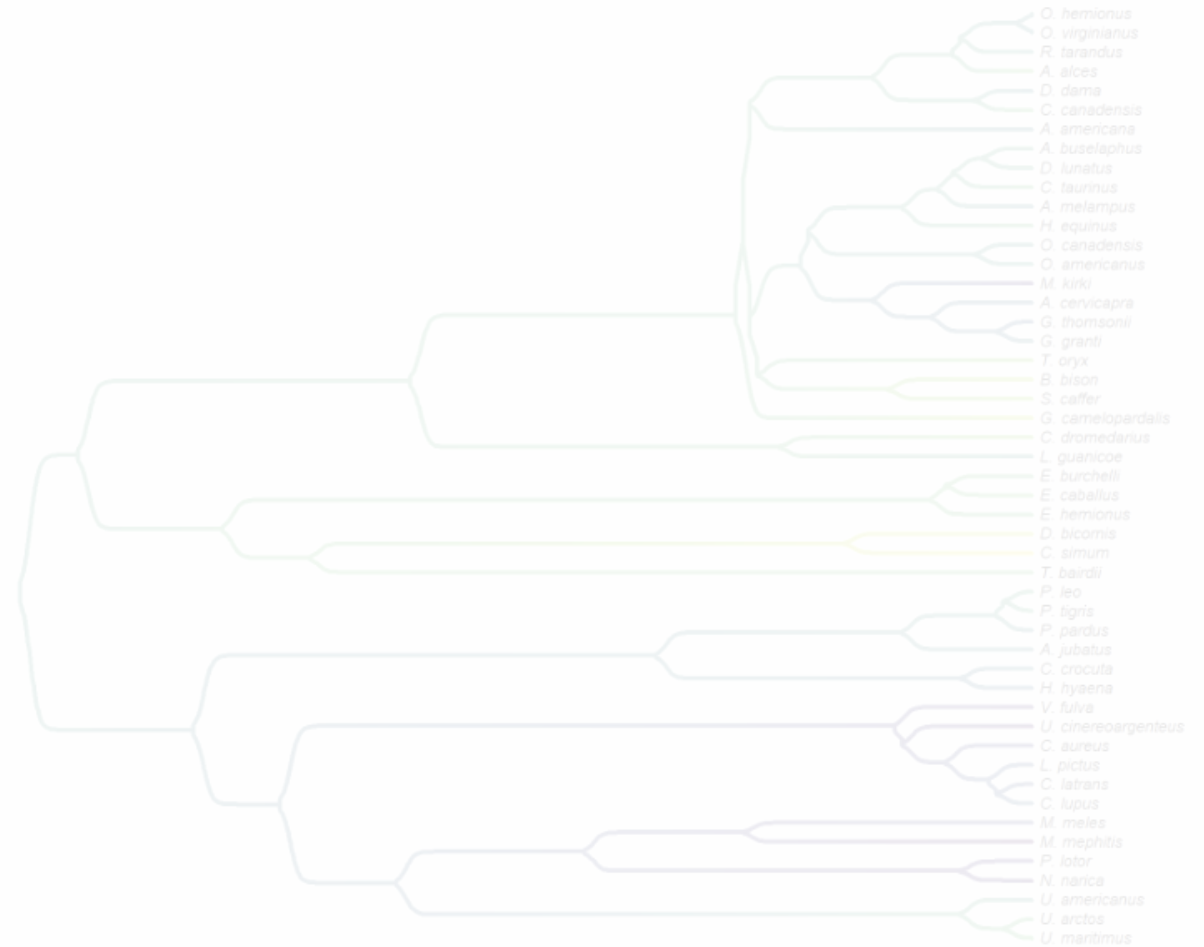
# Review of terminology



■ “homoplasy” – independently derived similarity.

# Inferring Phylogenies: Parsimony

- How do we figure out how many evolutionary changes are implied by a particular data pattern on the tree?



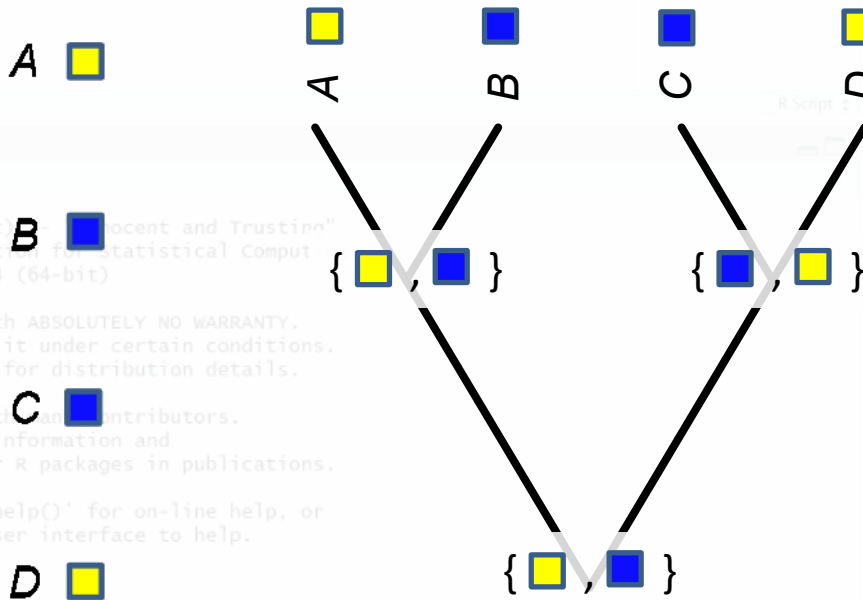
# Inferring Phylogenies: Parsimony

- The first *algorithm* that was described to compute the minimum number of changes implied by a data pattern on a tree is called the Fitch algorithm because it was identified by Walter Fitch (1929-2011).
- To apply the Fitch algorithm we start at the tips of the tree, and then we descend rootward (post-order) traversal through internal nodes.
- For each node, we compute the intersection between the sets for the nodes above. If the set is empty, we compute the union and +1 to the parsimony score.

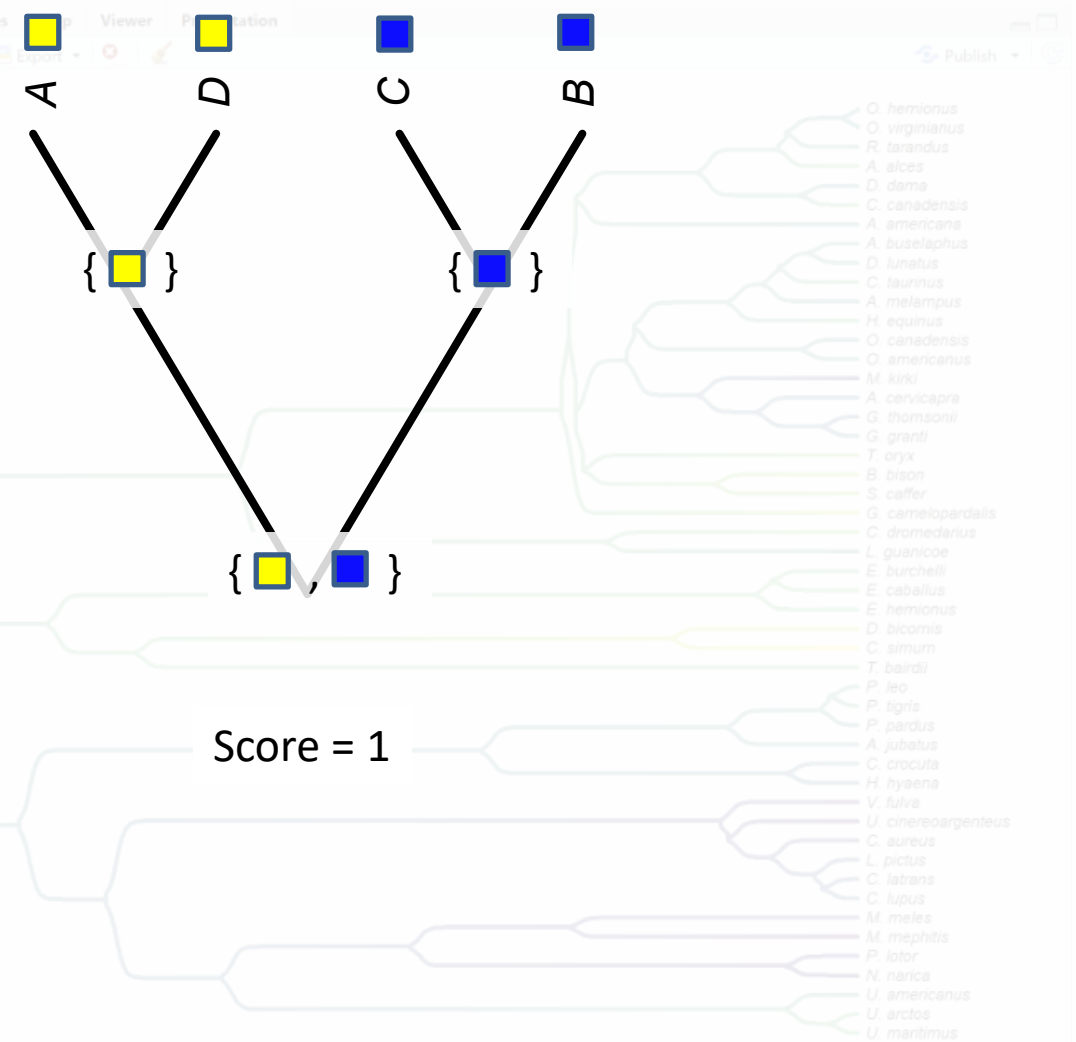




# Fitch algorithm



Score = 2



Score = 1

```
1 library(phytools)
2 data(mammal.tree)
3 data(mammal.data)
4 head(mammal.data)
5 lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
6 obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
7 obj<-setMap(obj, viridisLite::viridis(n=8))
8 sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
9
```

9.1 [Top Level]

Console Terminal Background Jobs

R 4.2.2 - /

R version 4.2.2 (2022-10-31 ucrt)  
Copyright (C) 2022 The R Foundation for Statistical Computing  
Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

```
> library(phytools)
Loading required package: ape
Loading required package: maps
> data(mammal.tree)
> data(mammal.data)
> head(mammal.data)
  bodyMass homeRange
U._maritimus 265.0    115.60
U._arctos    251.3     82.80
U._americanus 93.4     56.80
N._narica     4.4      1.05
P._lotor      7.0      1.14
M._mephitis   2.5      2.50
> lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
> obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
> obj<-setMap(obj, viridisLite::viridis(n=8))
> sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
> sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
>
```

Environment History Connections Tutorial

Import Dataset 390 MiB

mammal.out

mammal.tree

obj

Values

lnBodyMass

49 obs. of 2 variables

List of 5

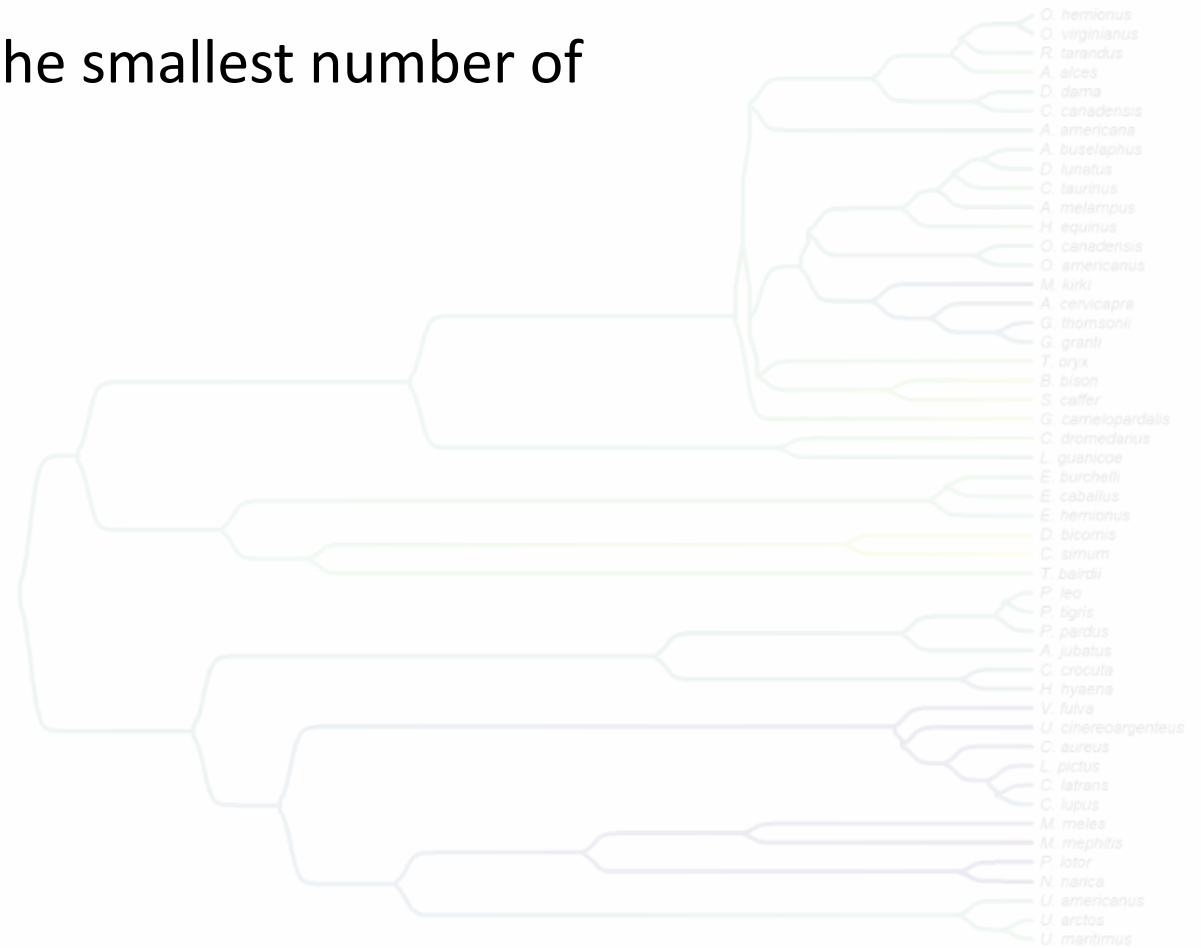
Large contMap (3 elements, 743.7 kb)

Named num [1:49] 5.58 5.53 4.54 1.48 1.95 ...

Q. hemionus  
Q. virginianus  
R. tarandus  
A. alces  
D. dama  
C. canadensis  
A. americana  
A. buselaphus  
D. lunatus  
C. taurinus  
A. melampus  
H. equinus  
O. canadensis  
O. americanus  
M. kirkii  
A. cervicapra  
G. thomsonii  
G. granti  
T. oryx  
B. bison  
S. caffer  
G. camelopardalis  
O. dromedarius  
L. guanicoe  
E. burchelli  
E. caballus  
E. hemionus  
D. bicornis  
C. simum  
T. bairdi  
P. leo  
P. tigris  
P. pardus  
A. jubatus  
C. crocuta  
H. hyena  
V. fulva  
U. cinereoargenteus  
C. aureus  
L. pictus  
C. latrans  
C. lupus  
M. meles  
M. mephitis  
P. lotor  
N. narica  
U. americanus  
U. arctos  
U. maritimus

# Inferring Phylogenies: Parsimony

- How do we figure out how many evolutionary changes are implied by a particular data pattern on the tree?
- How do we find the tree with the smallest number of evolutionary changes?



# Too many trees. . . .

Species (rooted)	Species (unrooted)	Number of trees
1	2	1
2	3	1
3	4	3
4	5	15
5	6	105
6	7	945
7	8	10,395
8	9	135,135
9	10	2,027,025
10	11	34,459,425
11	12	654,729,075
12	13	13,749,310,575
13	14	316,234,143,225
14	15	7,905,853,580,625
15	16	213,458,046,676,875
16	17	6,190,283,353,629,375
17	18	191,898,783,962,510,625
18	19	6,332,659,870,762,850,875
19	20	221,643,095,476,699,771,875
20	21	8,200,794,532,637,891,559,375
50	49	$2.753 \times 10^{76}$

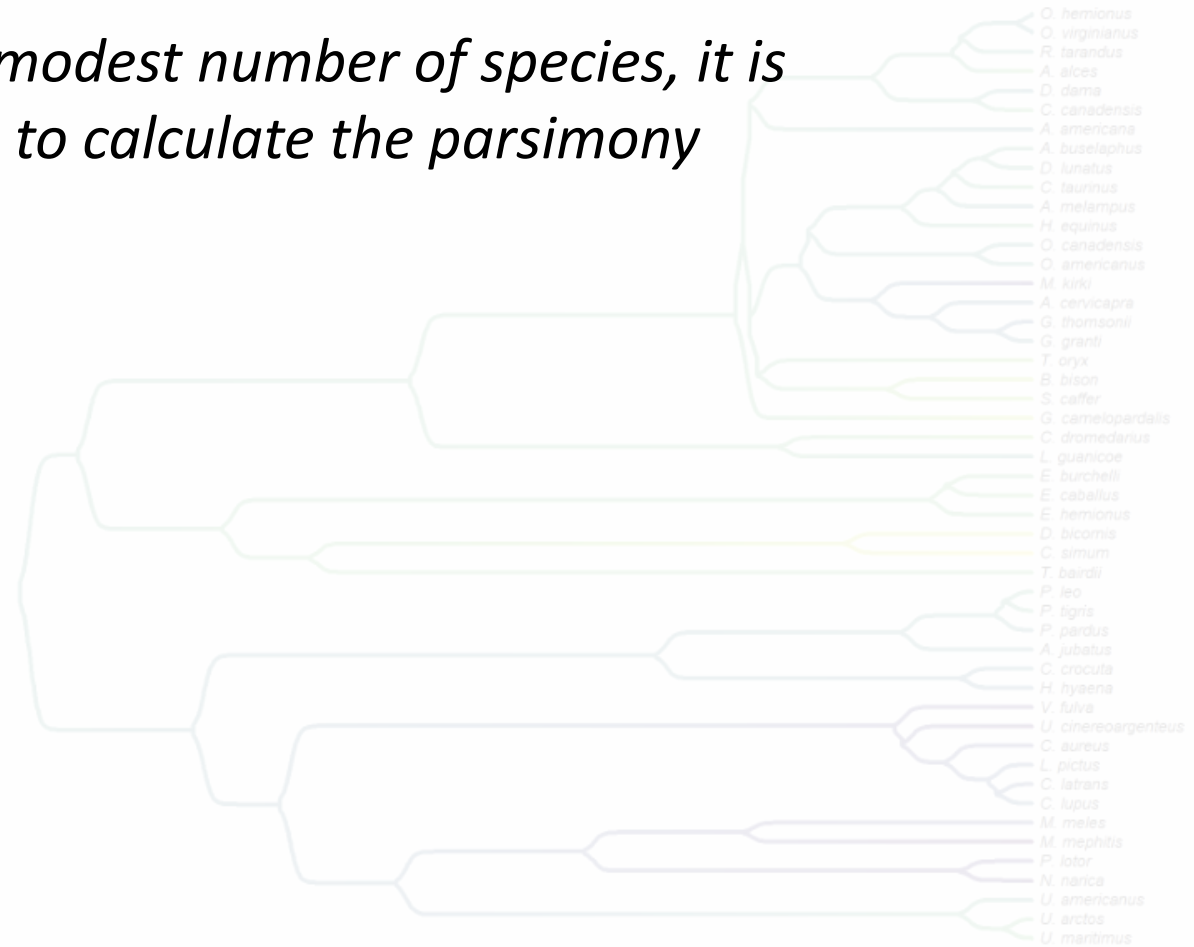
Number of electrons in the  
universe

$\sim 10^{79}$

Felsenstein, 2004

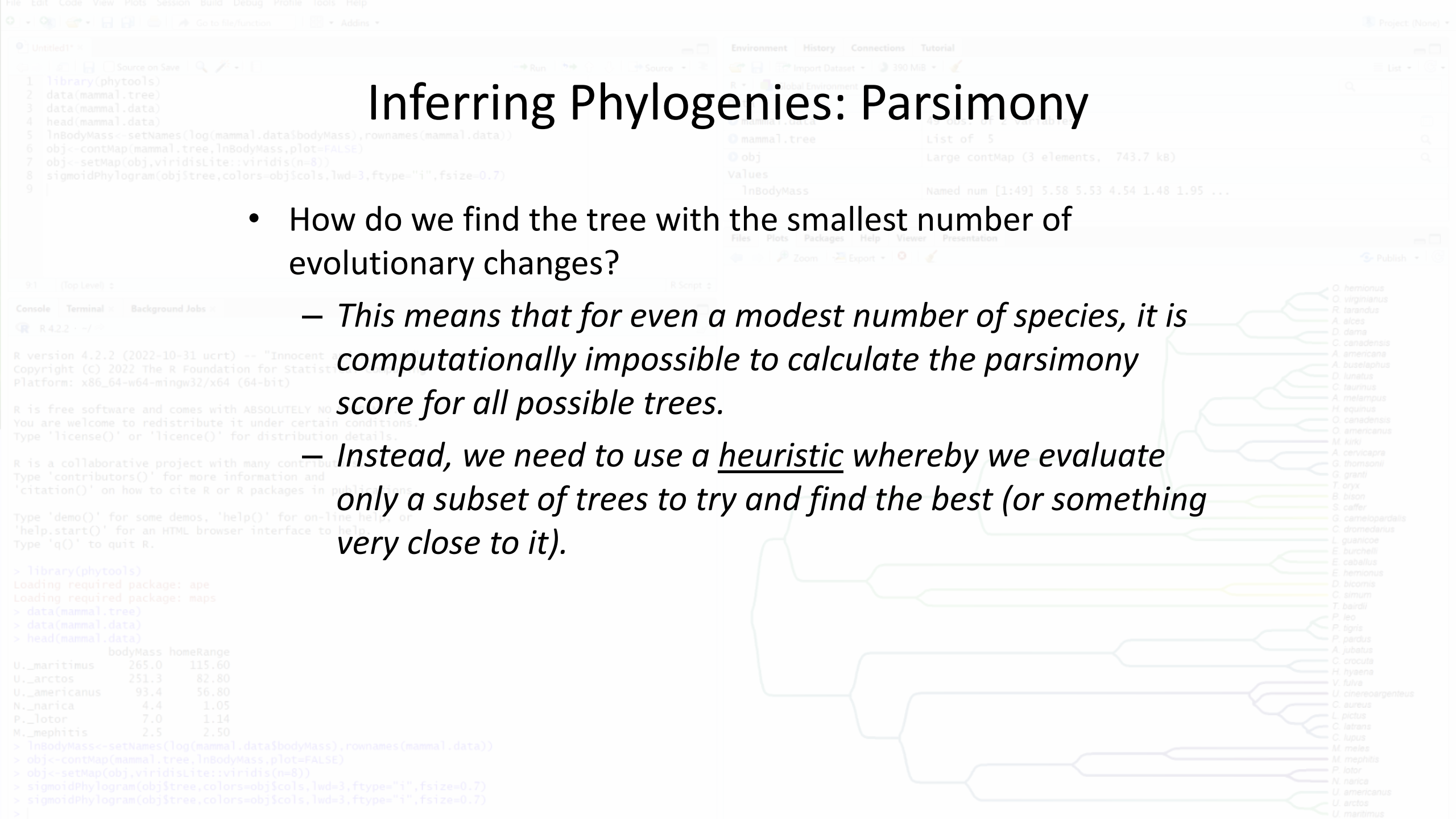
# Inferring Phylogenies: Parsimony

- How do we find the tree with the smallest number of evolutionary changes?
  - *This means that for even a modest number of species, it is computationally impossible to calculate the parsimony score for all possible trees.*



# Inferring Phylogenies: Parsimony

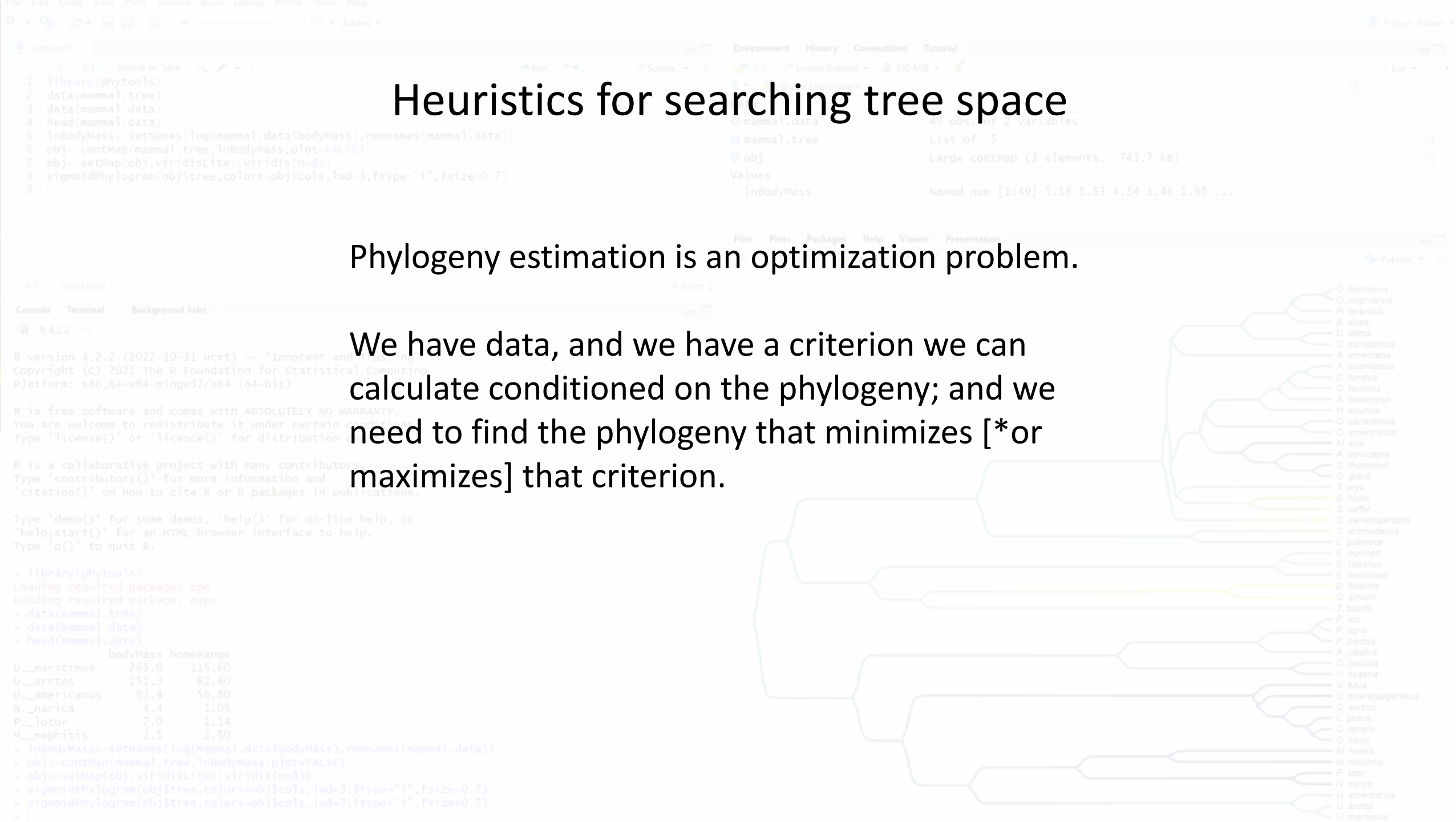
- How do we find the tree with the smallest number of evolutionary changes?
  - *This means that for even a modest number of species, it is computationally impossible to calculate the parsimony score for all possible trees.*
  - *Instead, we need to use a heuristic whereby we evaluate only a subset of trees to try and find the best (or something very close to it).*



# Heuristics for searching tree space

Phylogeny estimation is an optimization problem.

We have data, and we have a criterion we can calculate conditioned on the phylogeny; and we need to find the phylogeny that minimizes [\*or maximizes] that criterion.





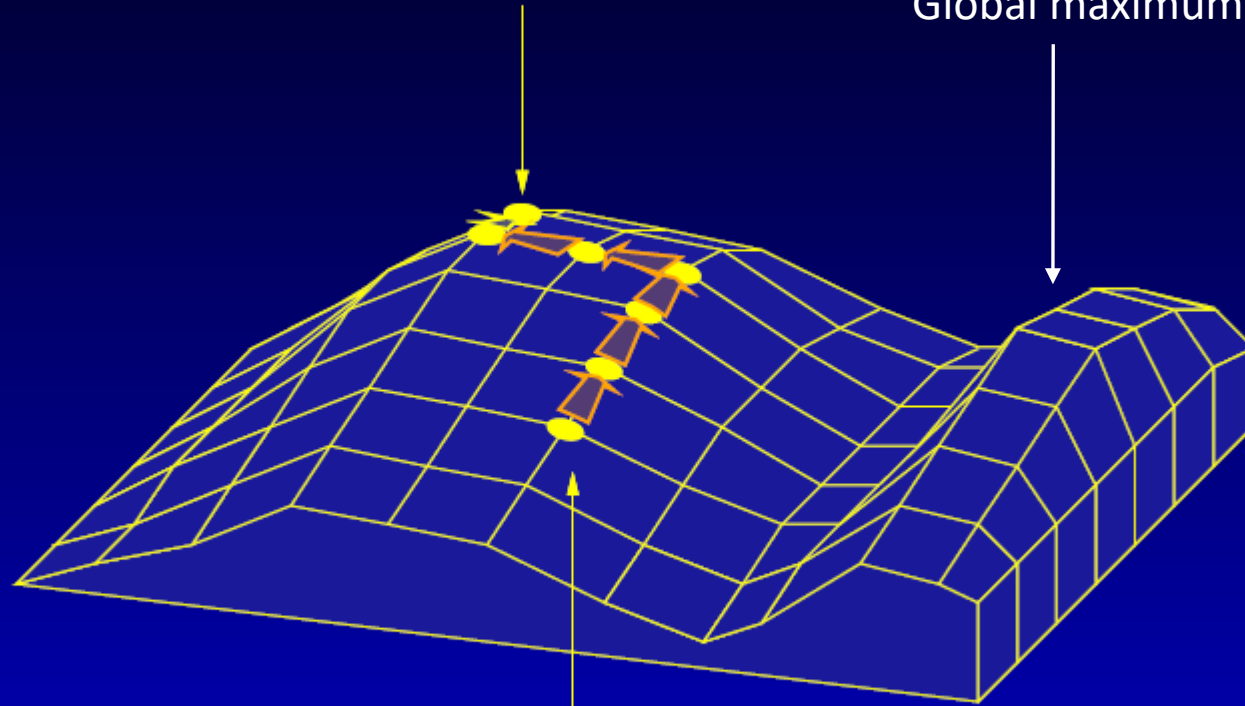
# Maximizing/minimizing optimality criteria

Greedy algorithm

end up here

Global maximum

If start here

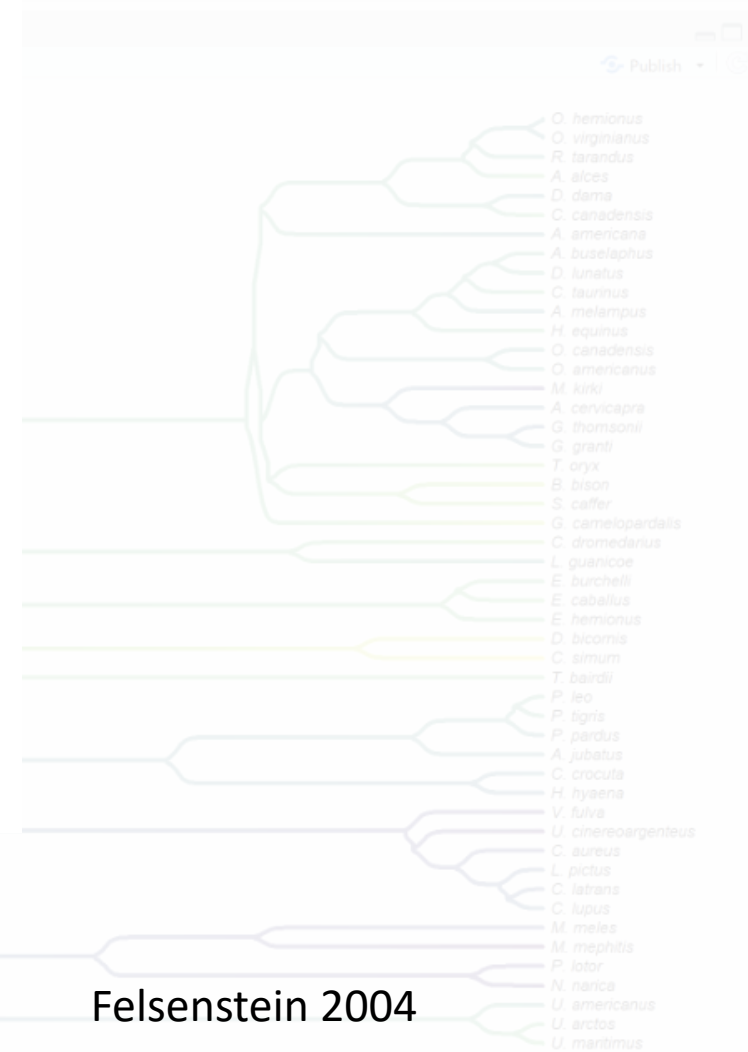
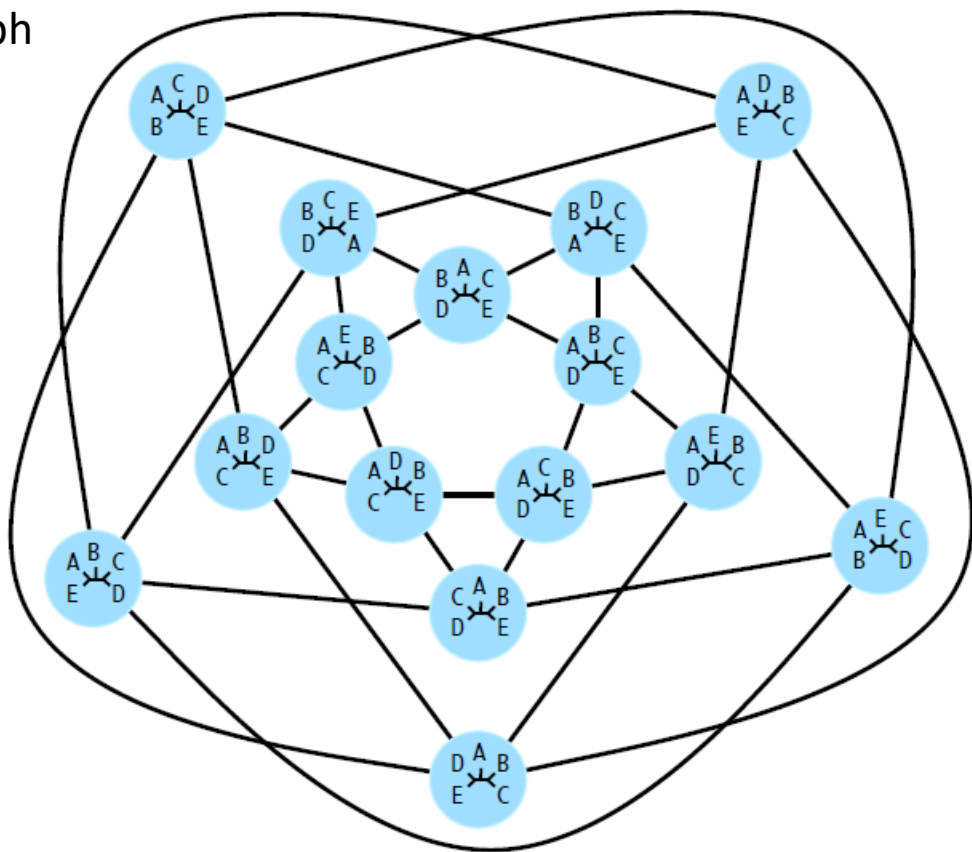


Felsenstein 2004

Phylogenetic tree showing relationships between various mammal species, including O. hemionus, O. virginianus, R. tarandus, A. alces, D. dama, C. canadensis, A. americana, A. buselaphus, D. lunatus, C. taurinus, A. melampus, H. equinus, O. canadensis, O. americanus, M. kirkii, A. cervicapra, G. thomsoni, G. granti, T. oryx, B. bison, S. caffer, G. camelopardalis, C. dromedarius, L. guanicoe, E. burchelli, E. caballus, E. hemionus, D. bicornis, C. simum, T. bairdi, P. leo, P. tigris, P. pardus, A. jubatus, C. crocuta, H. hyaena, V. fulva, U. cinereoargenteus, C. aureus, L. pictus, C. latrans, C. lupus, M. meles, M. mephitis, P. lotor, N. narica, U. americanus, U. arctos, and U. maritimus.

# "Treespace"

Schoenberg graph



```
1 library(phytools)
2 data(mammal.tree)
3 data(mammal.data)
4 head(mammal.data)
5 lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
6 obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
7 obj<-setMap(obj, viridisLite::viridis(n=8))
8 sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
9
```

R 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"  
Copyright (C) 2022 The R Foundation for Statistical Computing  
Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

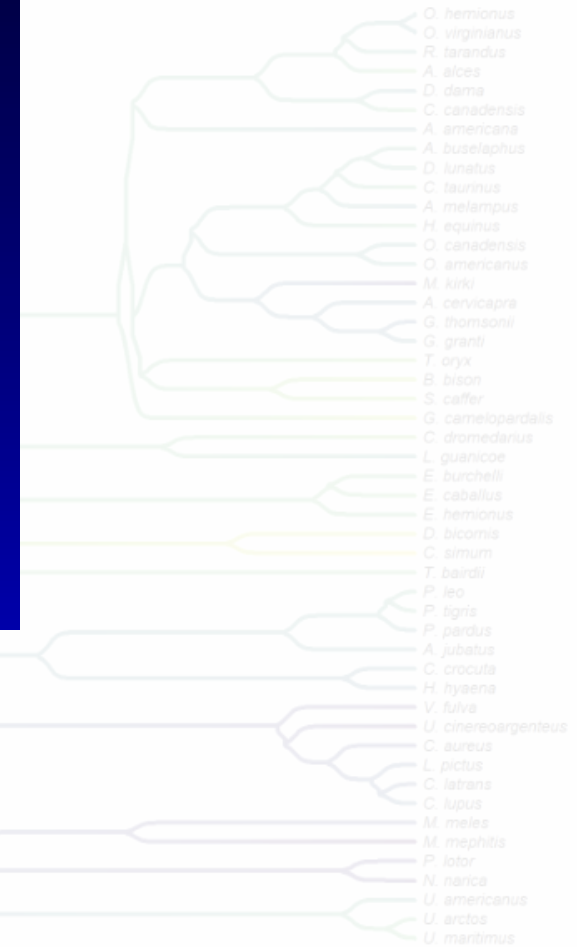
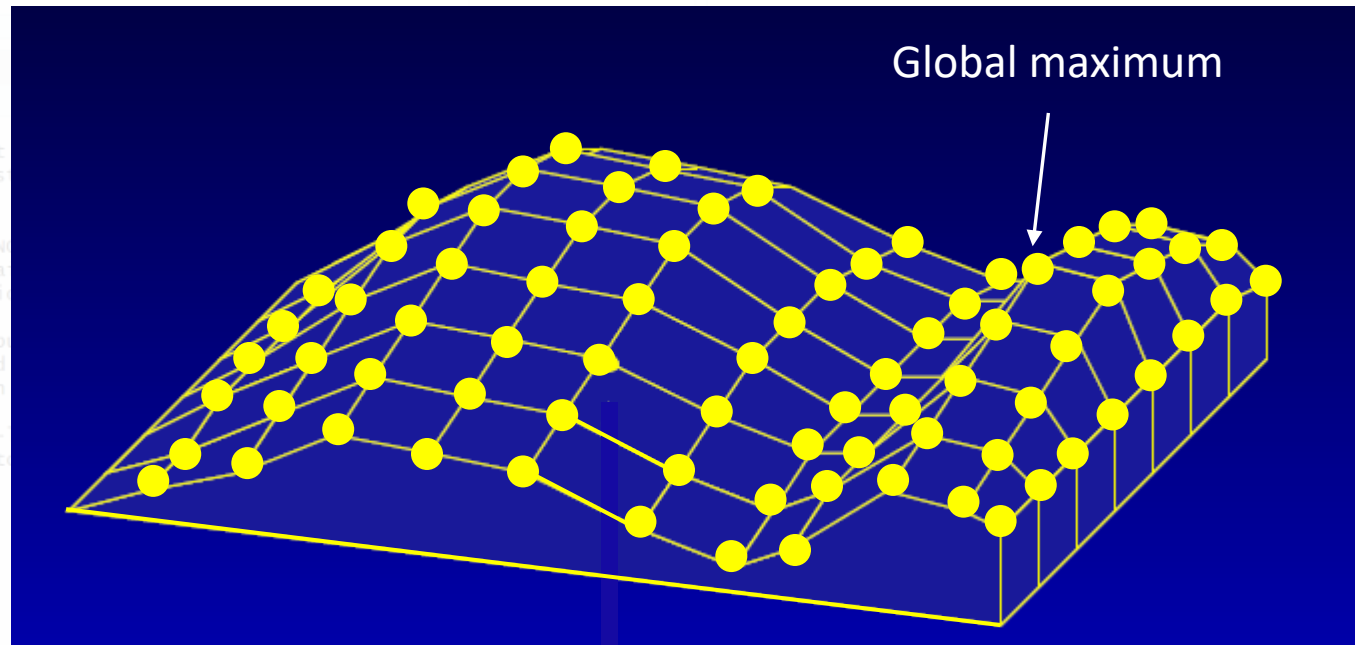
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

```
> library(phytools)
Loading required package: ape
Loading required package: maps
> data(mammal.tree)
> data(mammal.data)
> head(mammal.data)
      bodyMass homeRange
U._maritimus    265.0     115.60
U._arctos       251.3      82.80
U._americanus   93.4      56.80
N._narica        4.4       1.05
P._lotor         7.0       1.14
M._mephitis      2.5       2.50
> lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
> obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
> obj<-setMap(obj, viridisLite::viridis(n=8))
> sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
> sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
>
```

# Methods to explore treespace

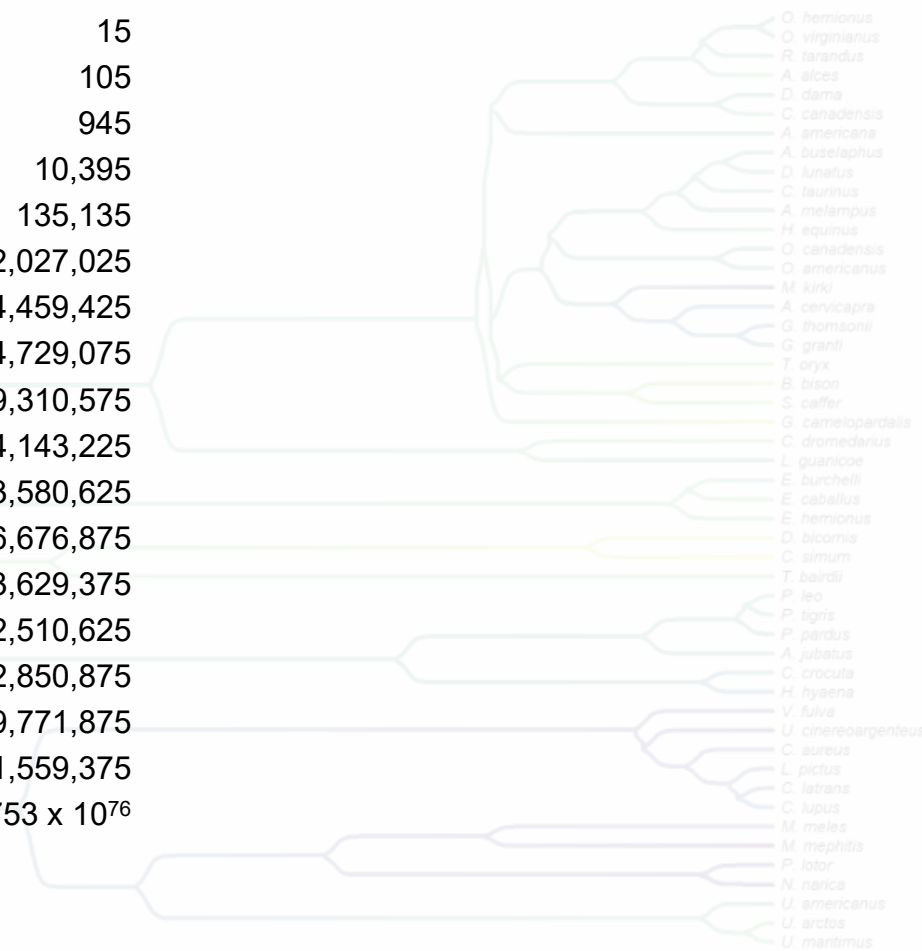
## 1. Exhaustive search: enumerate and evaluate all possible trees.



# Remember, too many trees. . . .

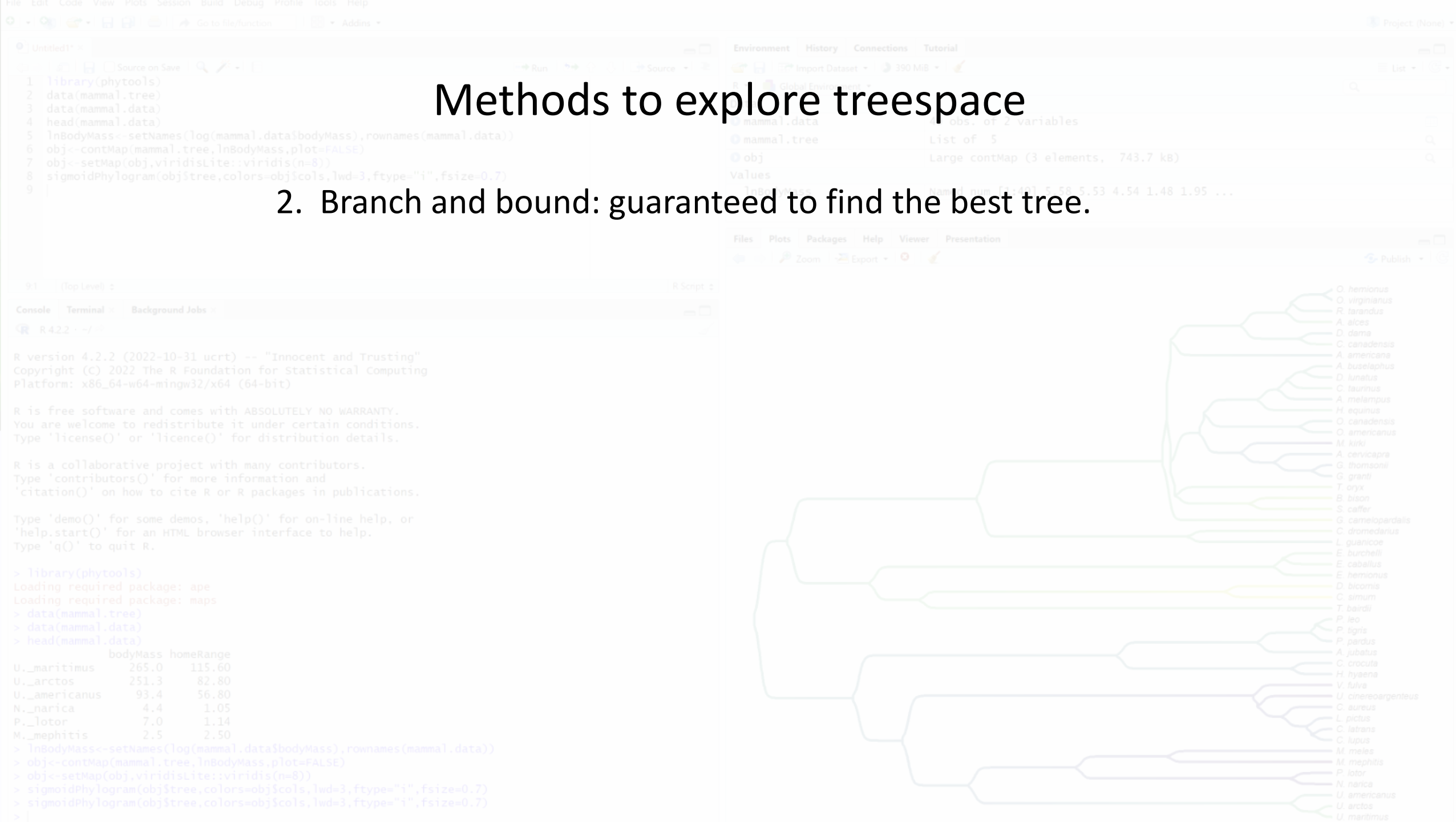
Species (rooted)	Species (unrooted)	Number of trees
1	2	1
2	3	1
3	4	3
4	5	15
5	6	105
6	7	945
7	8	10,395
8	9	135,135
9	10	2,027,025
10	11	34,459,425
11	12	654,729,075
12	13	13,749,310,575
13	14	316,234,143,225
14	15	7,905,853,580,625
15	16	213,458,046,676,875
16	17	6,190,283,353,629,375
17	18	191,898,783,962,510,625
18	19	6,332,659,870,762,850,875
19	20	221,643,095,476,699,771,875
20	21	8,200,794,532,637,891,559,375
50	51	$2.753 \times 10^{76}$

Felsenstein, 2004



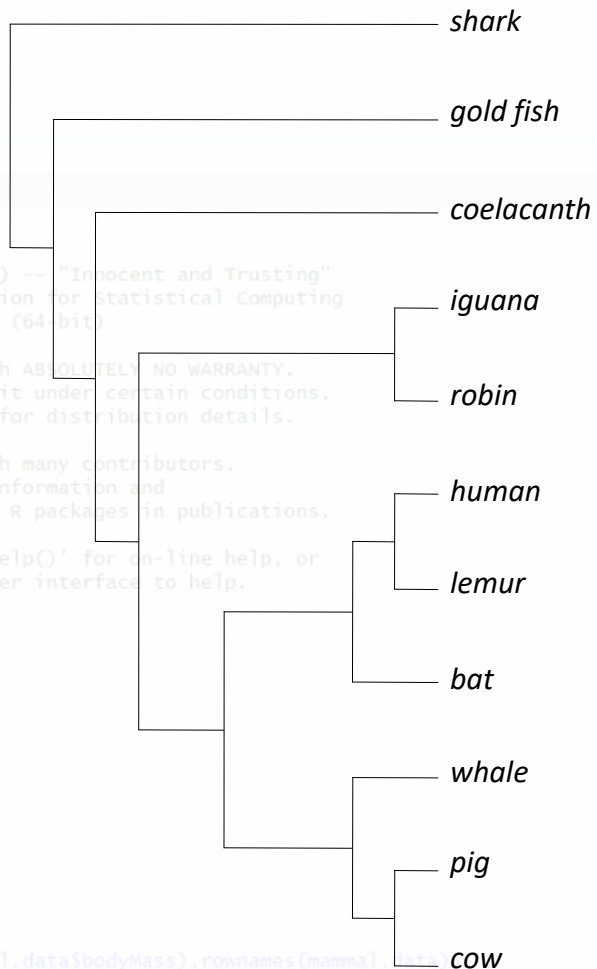
# Methods to explore treespace

## 2. Branch and bound: guaranteed to find the best tree.

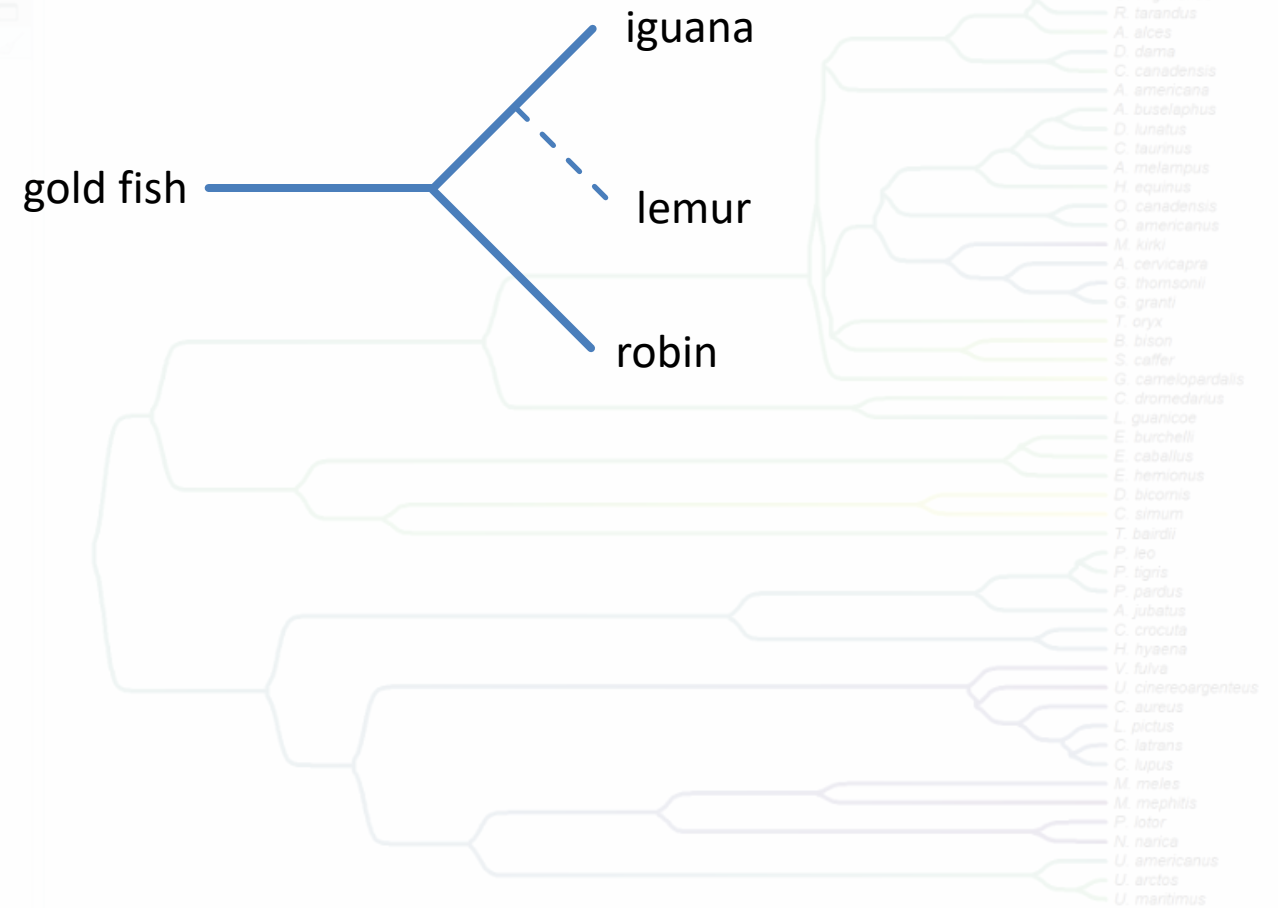


# Branch & bound algorithm

Score = 13



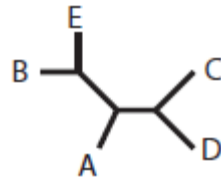
Score = 14



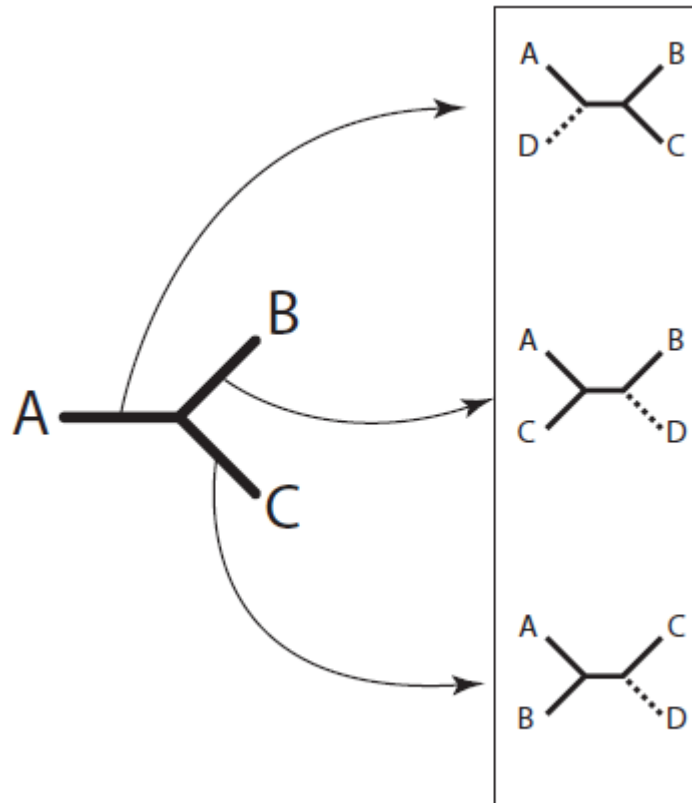


# Branch & bound algorithm

Starting tree



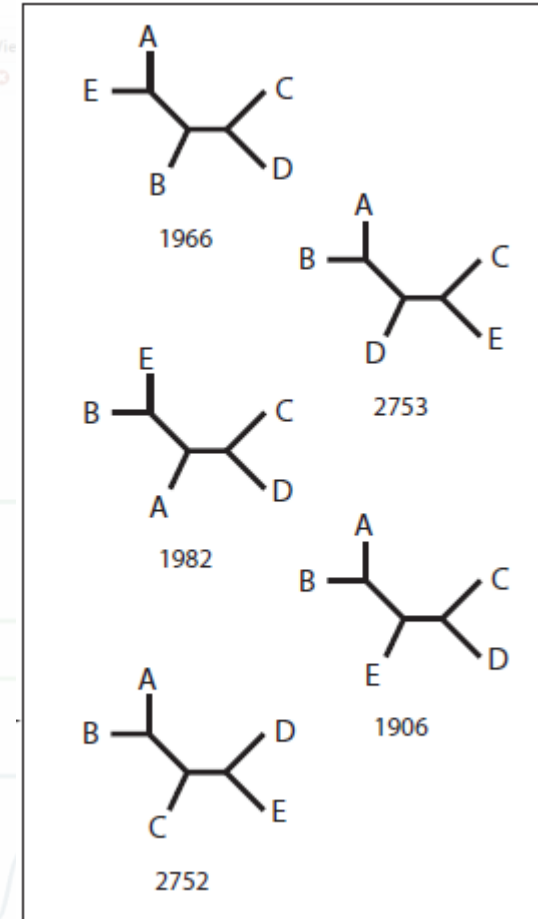
Score = 15



Score = 16

Score = 20

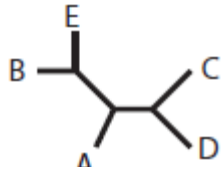
Score = 12



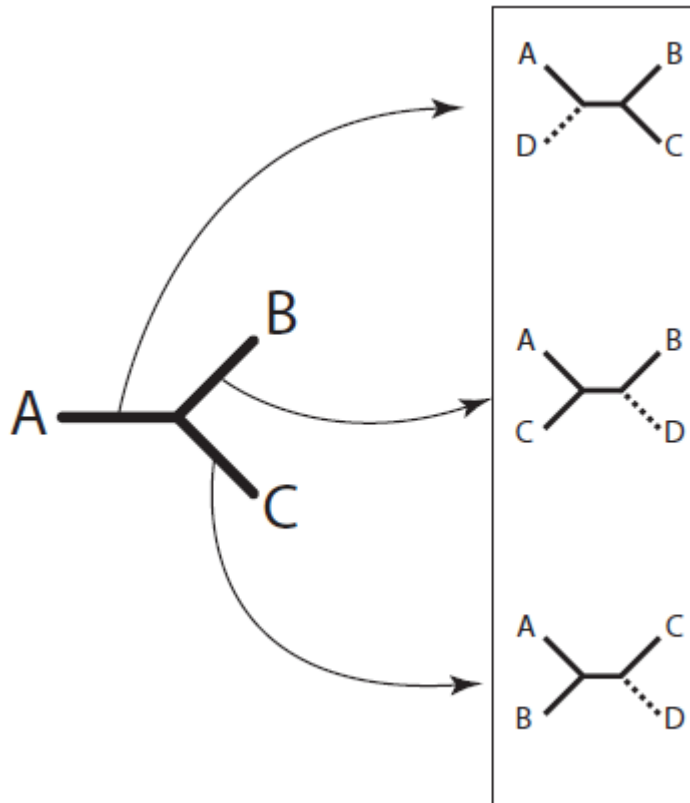
Modified from Holder 2011

# Branch & bound algorithm

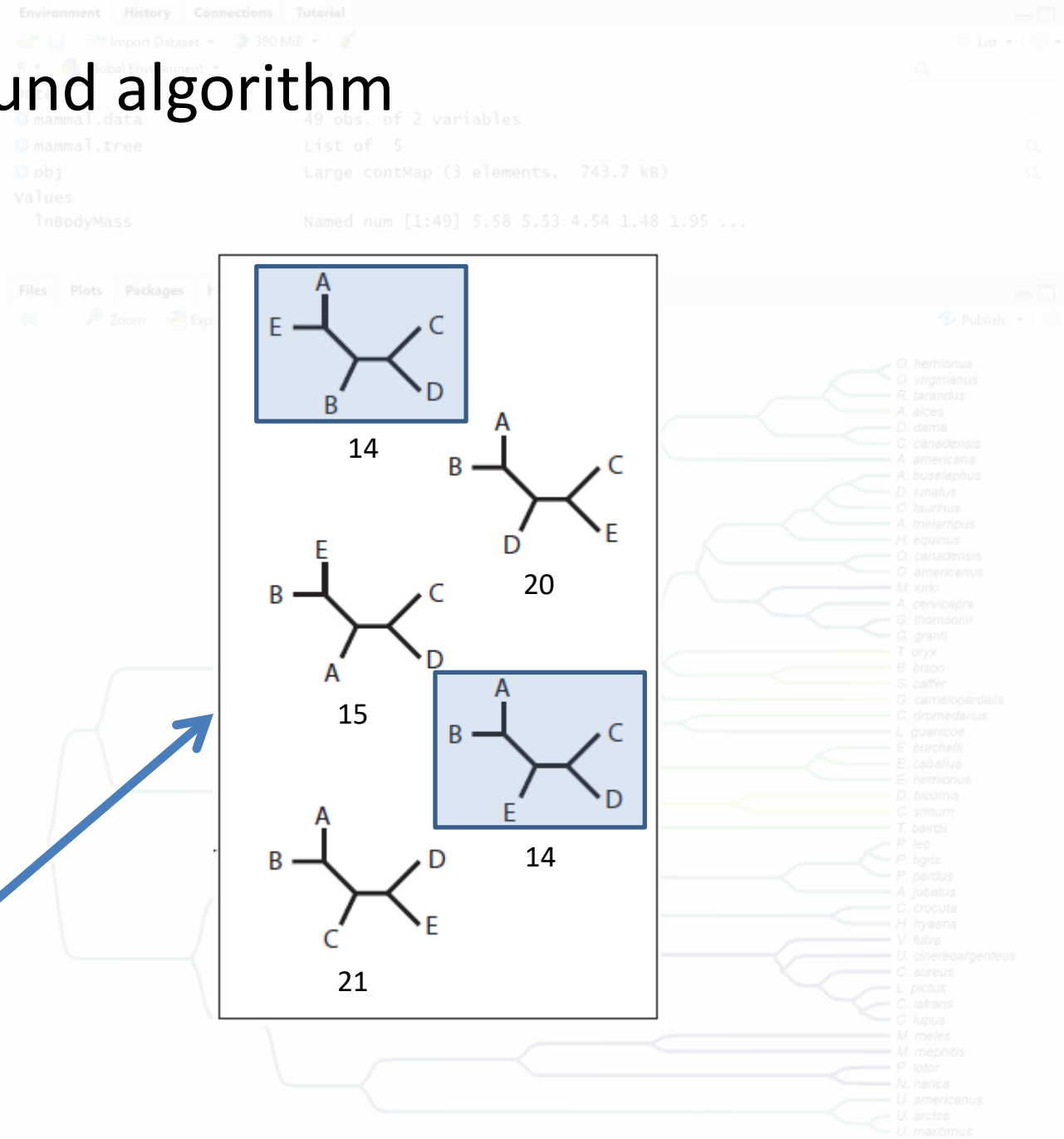
Starting tree



Score = 15

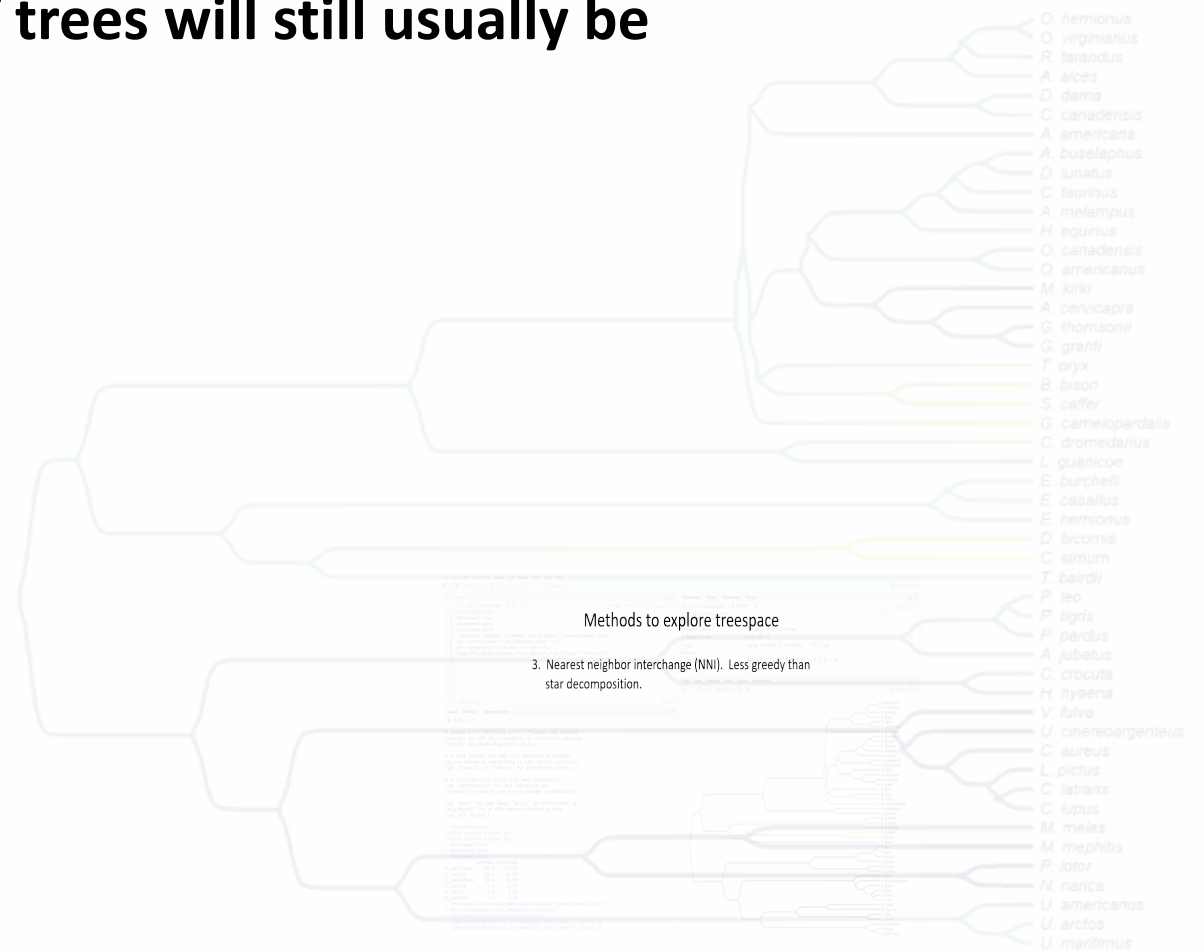


Modified from Holder 2011



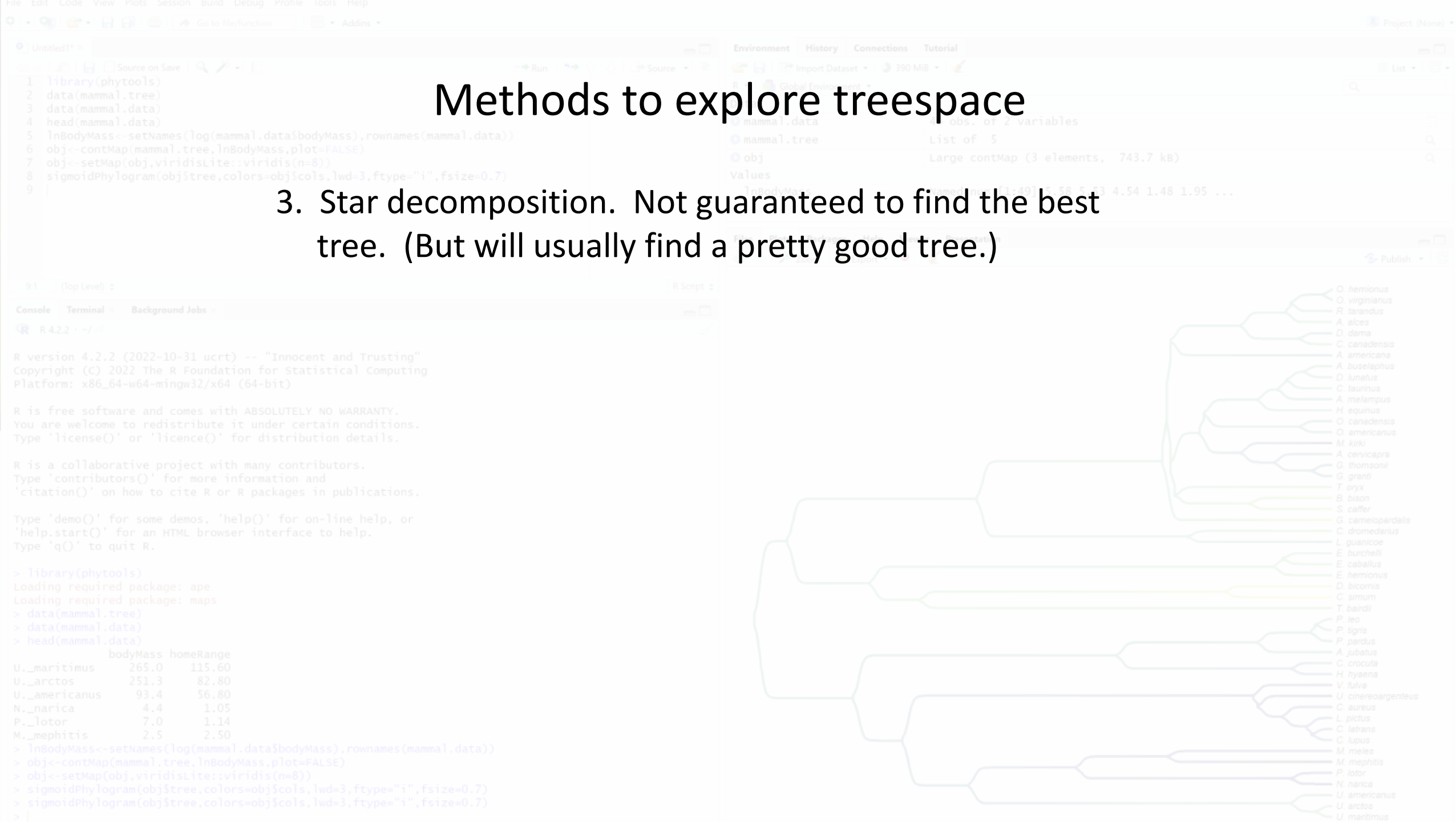
# Methods to explore treespace

2. Branch and bound: guaranteed to find the best tree (although it may not find *all* the best trees).  
*Furthermore*, the number of trees will still usually be too large.



# Methods to explore treespace

## 3. Star decomposition. Not guaranteed to find the best tree. (But will usually find a pretty good tree.)



# Star decomposition

```
1 library(phytools)
2 data(mammal.tree)
3 data(mammal.data)
4 head(mammal.data)
5 lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
6 obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
7 obj<-setMap(obj, viridisLite::viridis(n=8))
8 sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
9
```

9.1 (Top Level) :

Console Terminal Background Jobs

R 4.2.2 - / -

R version 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"  
Copyright (C) 2022 The R Foundation for Statistical Computing  
Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

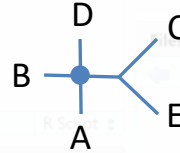
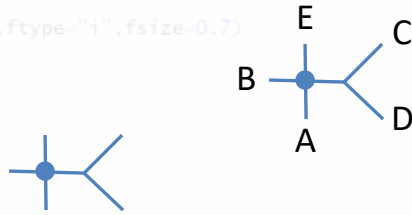
```
> library(phytools)
Loading required package: ape
Loading required package: maps
```

```
> data(mammal.tree)
> data(mammal.data)
> head(mammal.data)
```

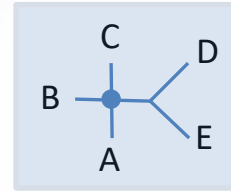
	bodyMass	homeRange
U._maritimus	265.0	115.60
U._arctos	251.3	82.80
U._americanus	93.4	56.80
N._narica	4.4	1.05
P._lotor	7.0	1.14
M._mephitis	2.5	2.50

```
> lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
> obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
> obj<-setMap(obj, viridisLite::viridis(n=8))
> sigmoidPhylogram(obj$tree, colors=obj$cols, lwd=3, ftype="i", fsize=0.7)
>
```

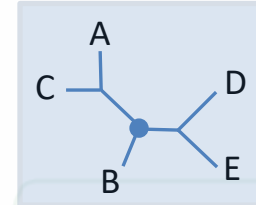
Modified from Holder 2011



BEST!



BEST!



# Methods to explore treespace

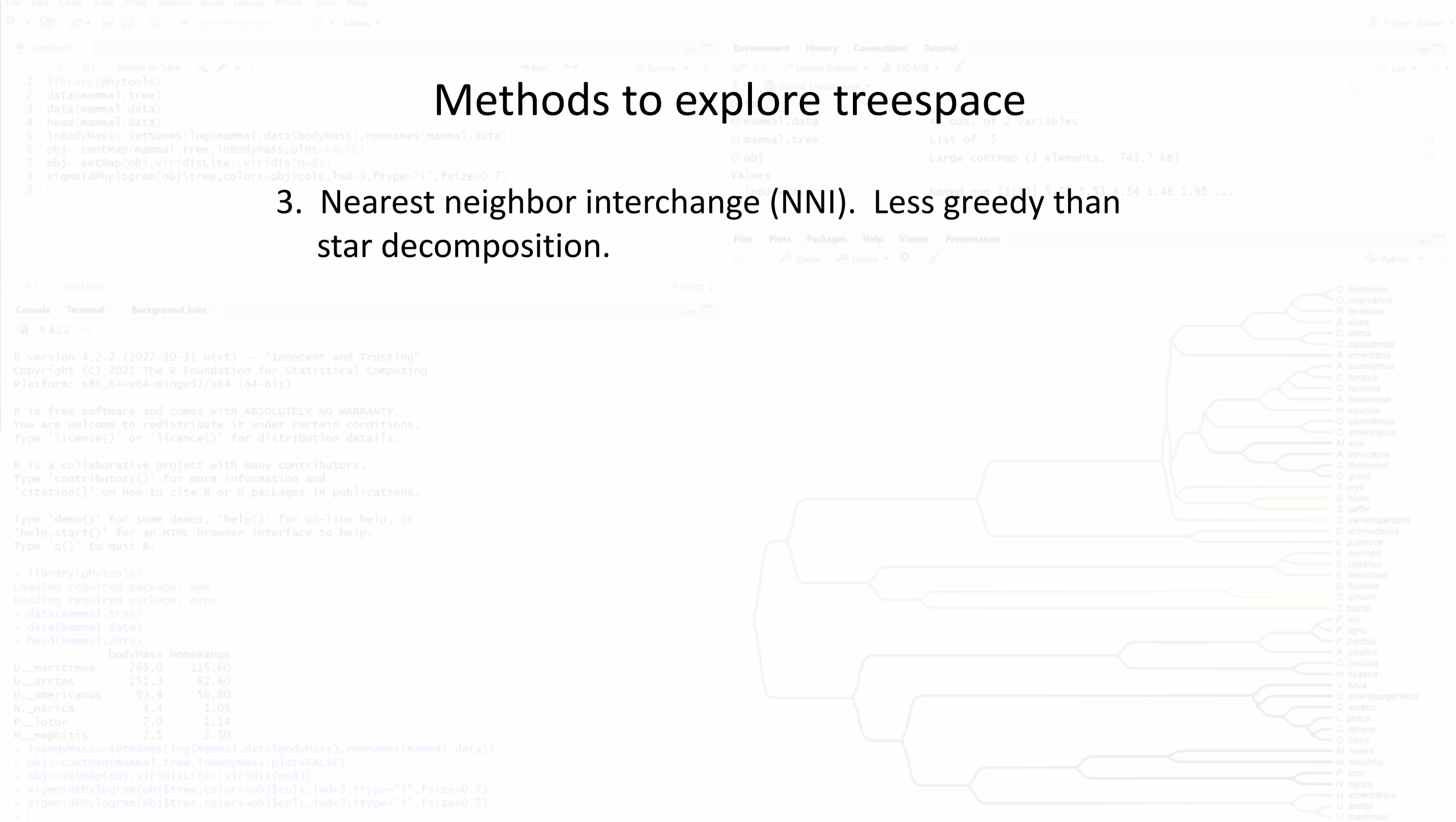
3. Star decomposition. Not guaranteed to find the best tree. (But will usually find a pretty good tree.) **Very greedy! Will usually fail to find the best tree. (This has been shown to be true even in datasets with no homoplasy!)**



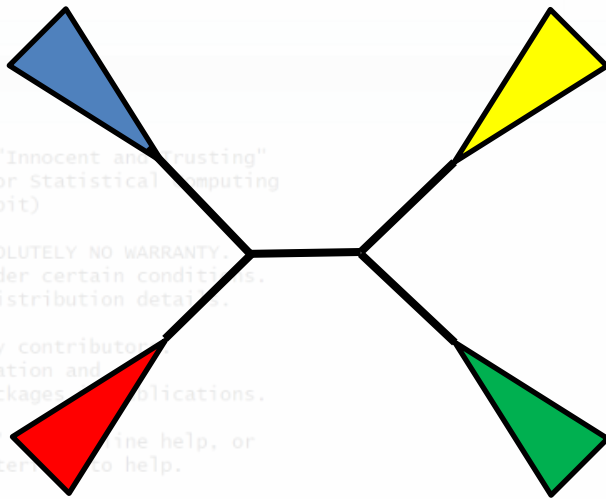


# Methods to explore treespace

## 3. Nearest neighbor interchange (NNI). Less greedy than star decomposition.

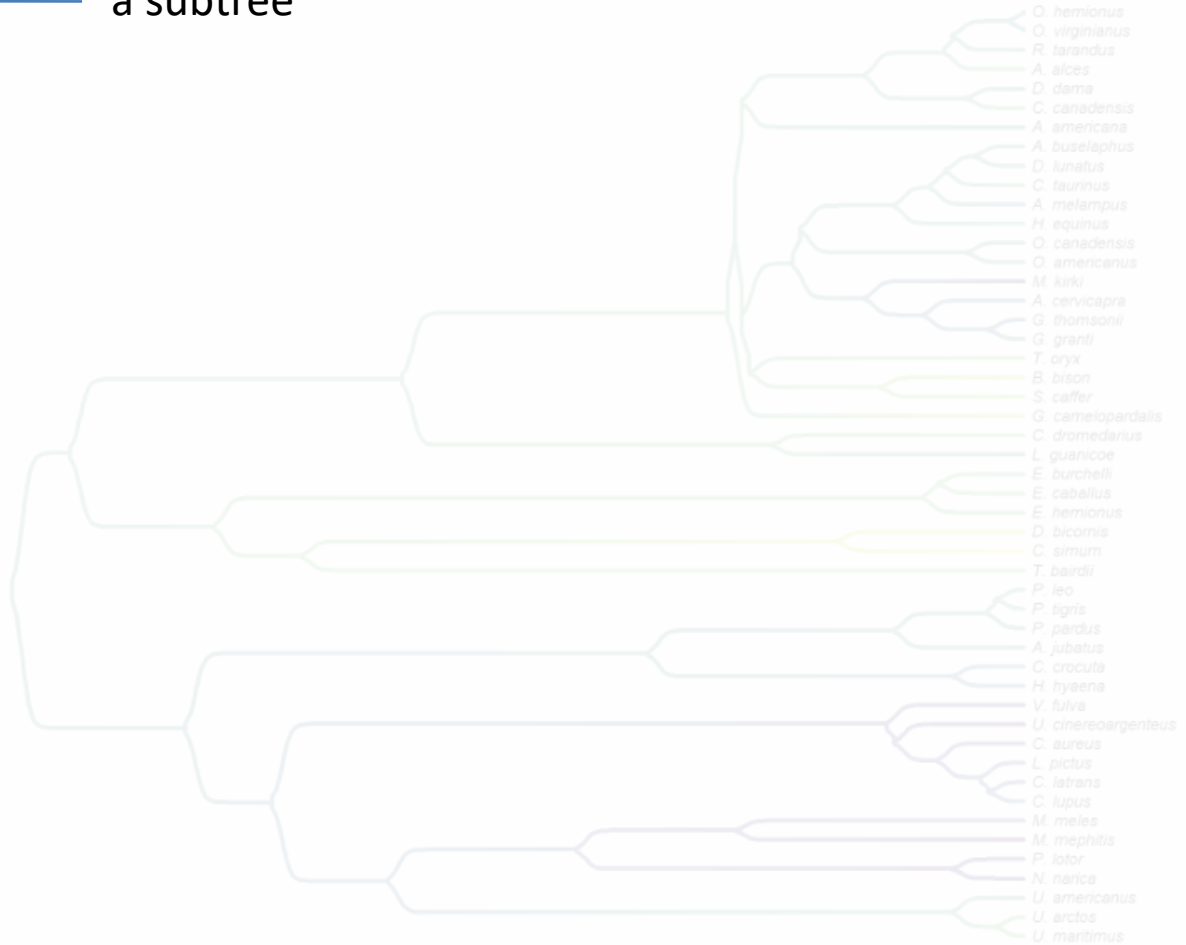
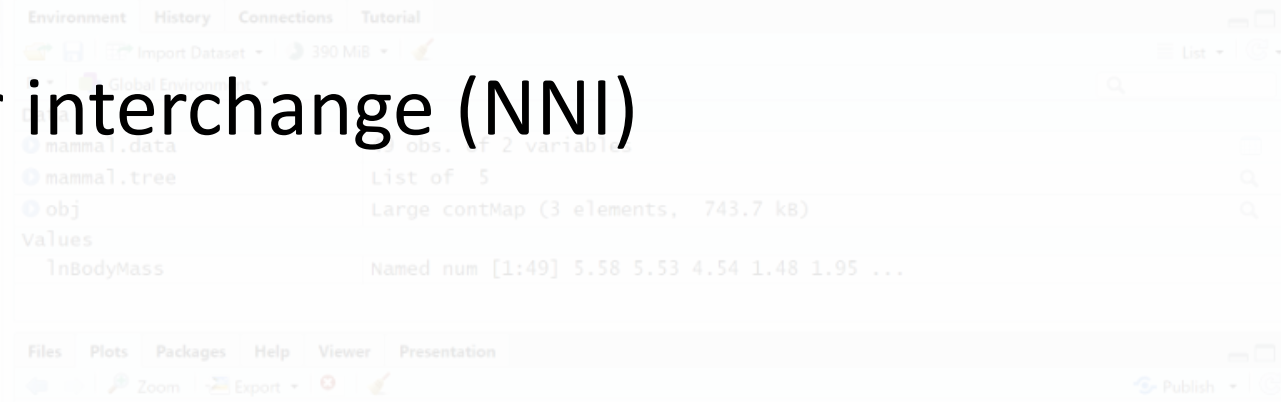


# Nearest neighbor interchange (NNI)



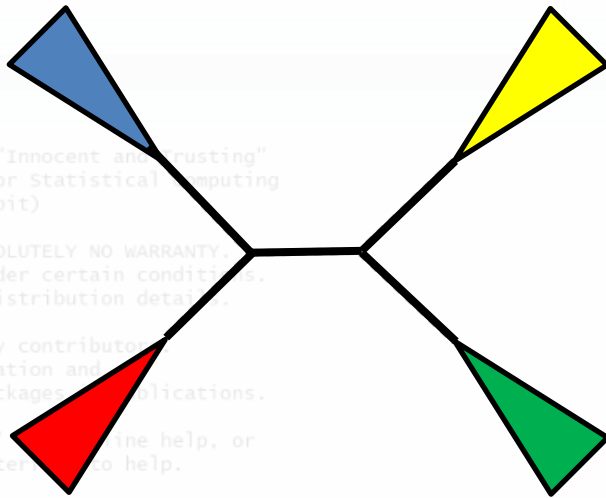
a subtree

Felsenstein 2004

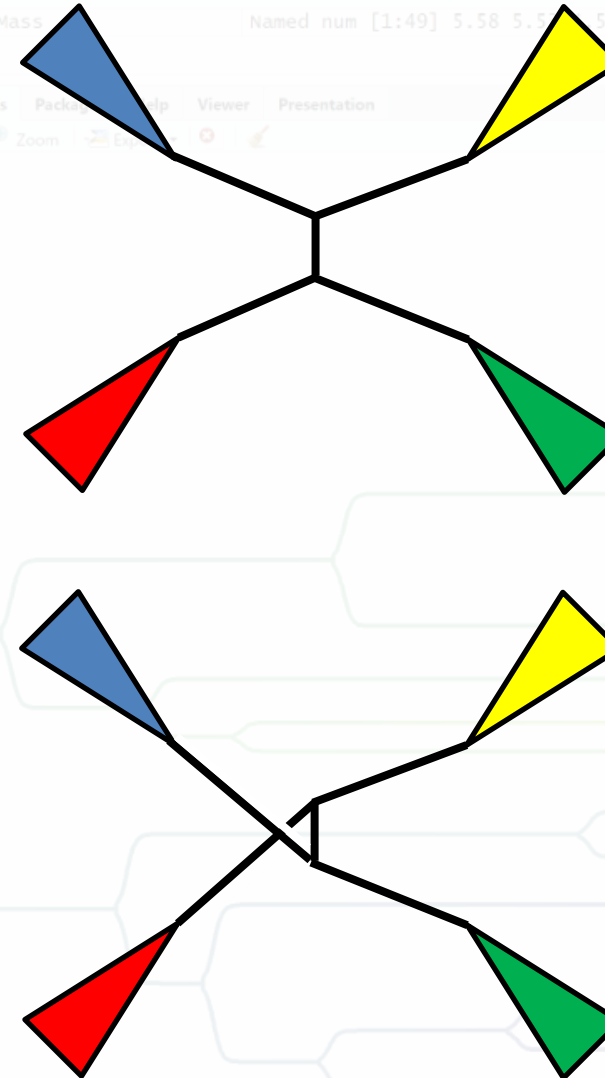


# Nearest neighbor interchange (NNI)

Dissolve internal branches



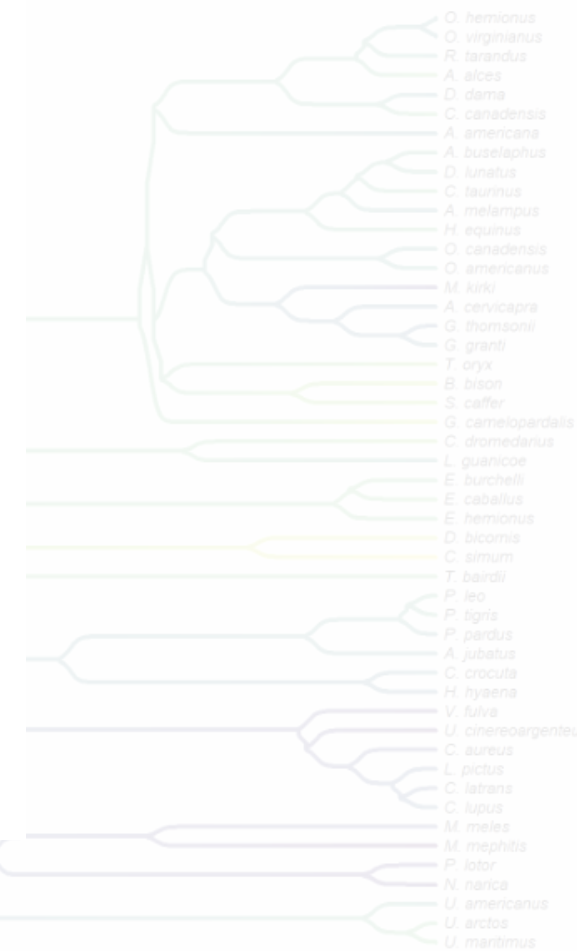
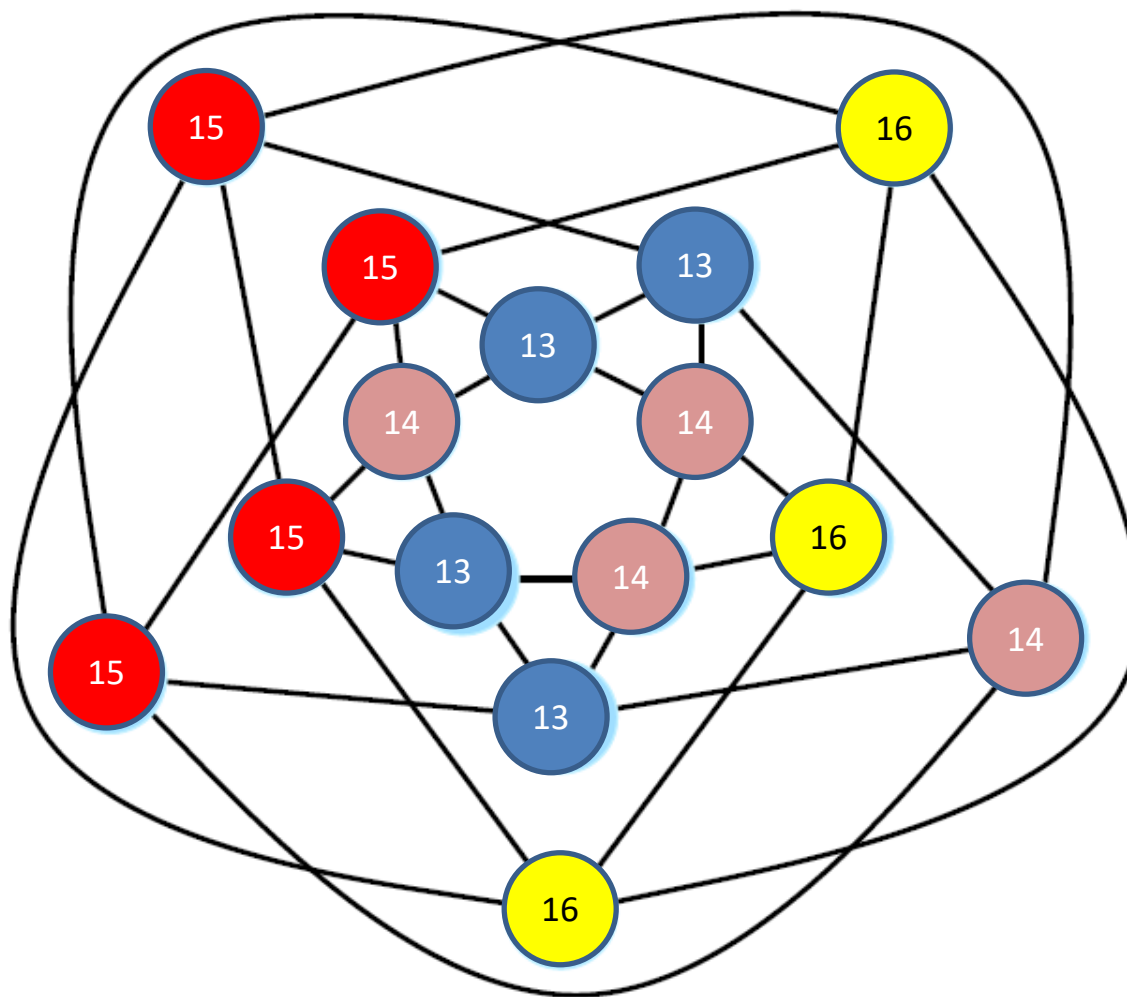
Reattach



Felsenstein 2004

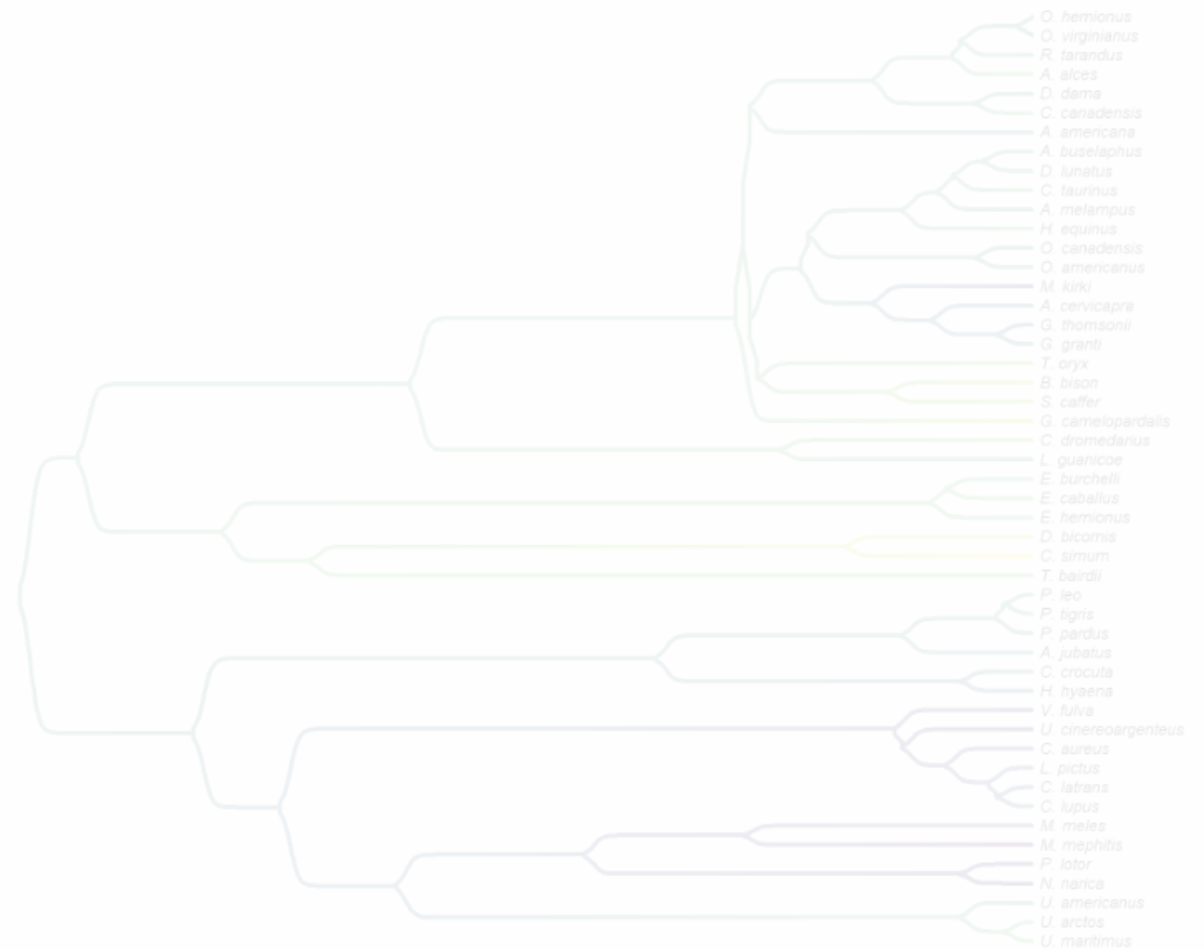
# Nearest neighbor interchange (NNI)

“Islands” in  
treespace



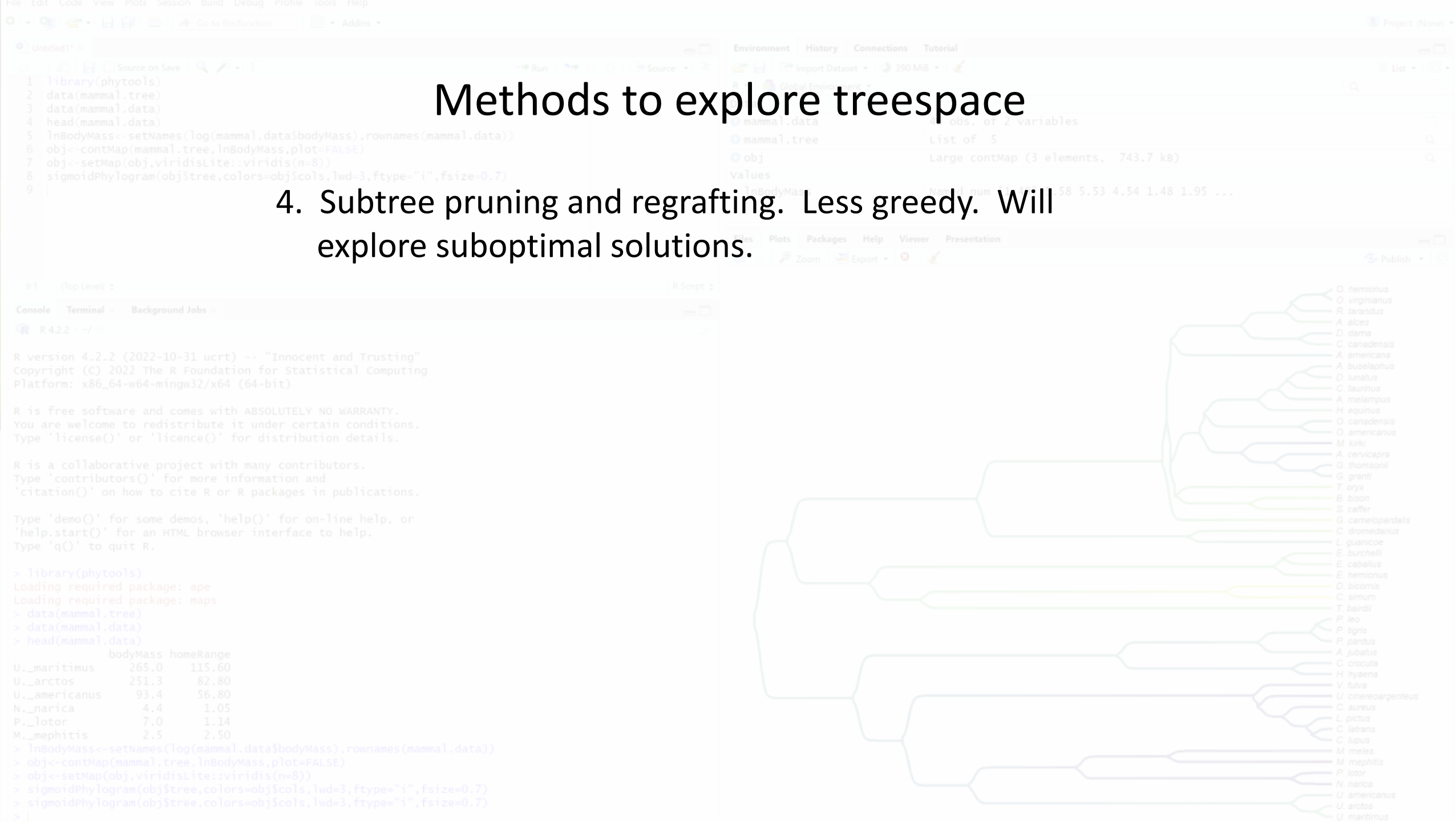
# Methods to explore treespace

3. Nearest neighbor interchange (NNI). Less greedy than star decomposition. **Can get trapped on local optima.**



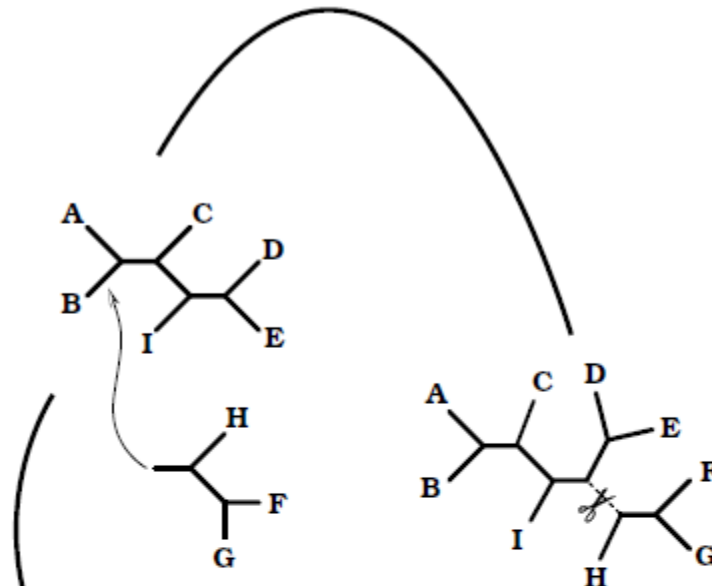
# Methods to explore treespace

## 4. Subtree pruning and regrafting. Less greedy. Will explore suboptimal solutions.

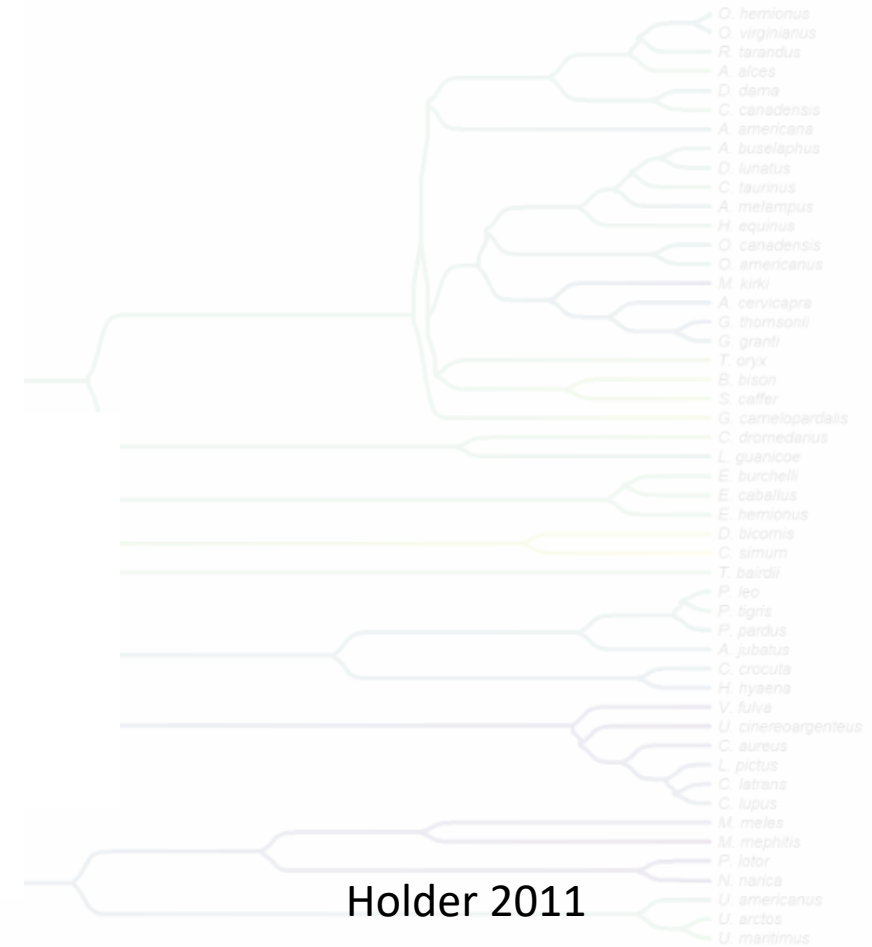




# Subtree pruning and regrafting (SPR)

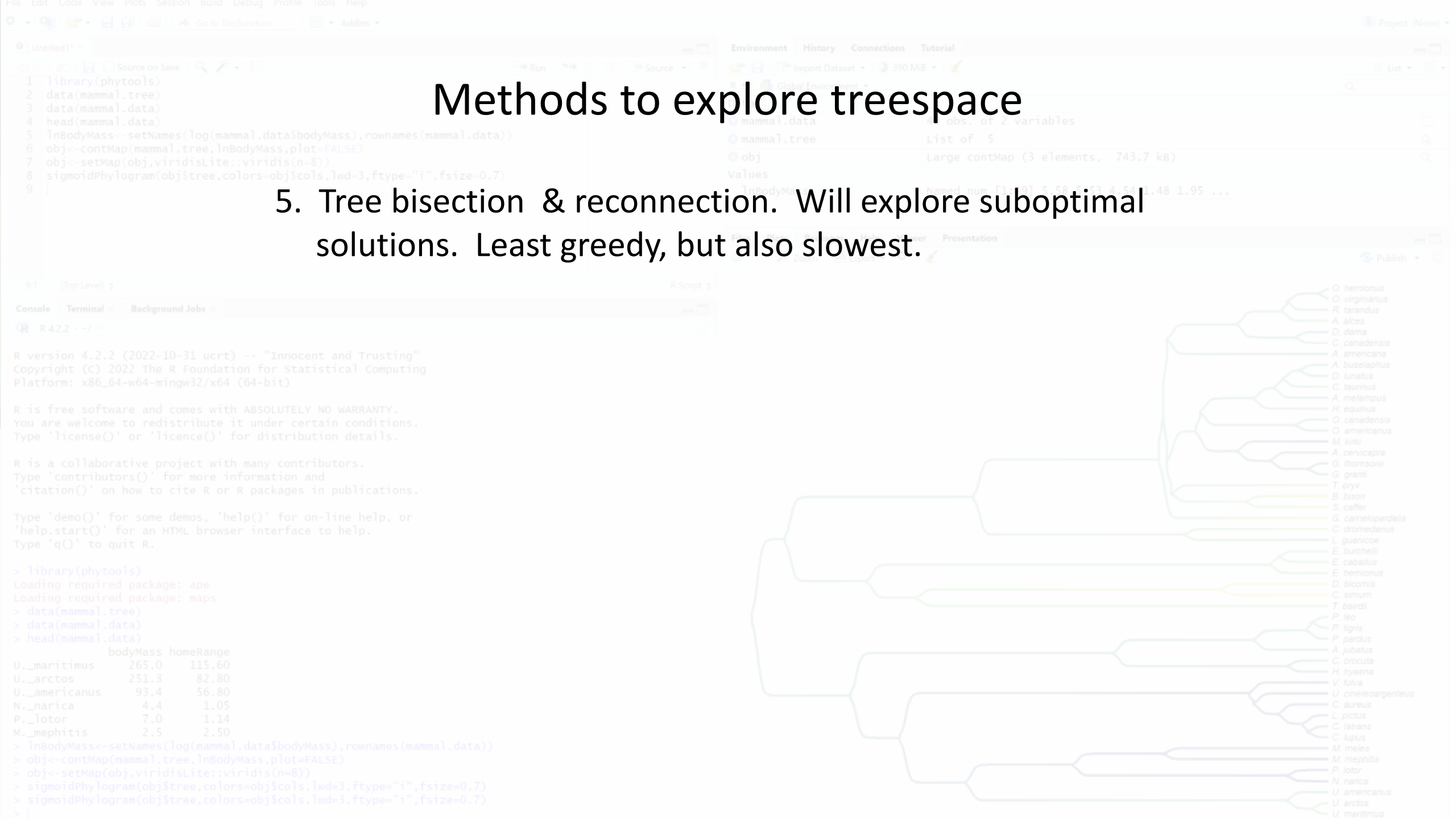


SPR maintains  
subtree rooting

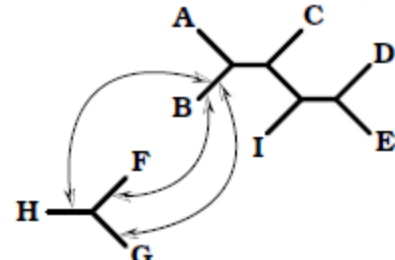
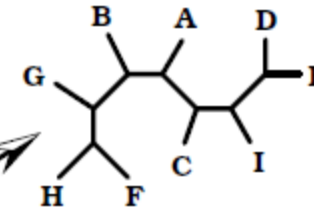
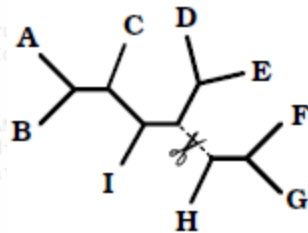
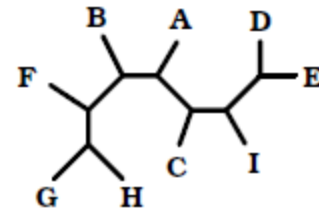


# Methods to explore treespace

5. Tree bisection & reconnection. Will explore suboptimal solutions. Least greedy, but also slowest.

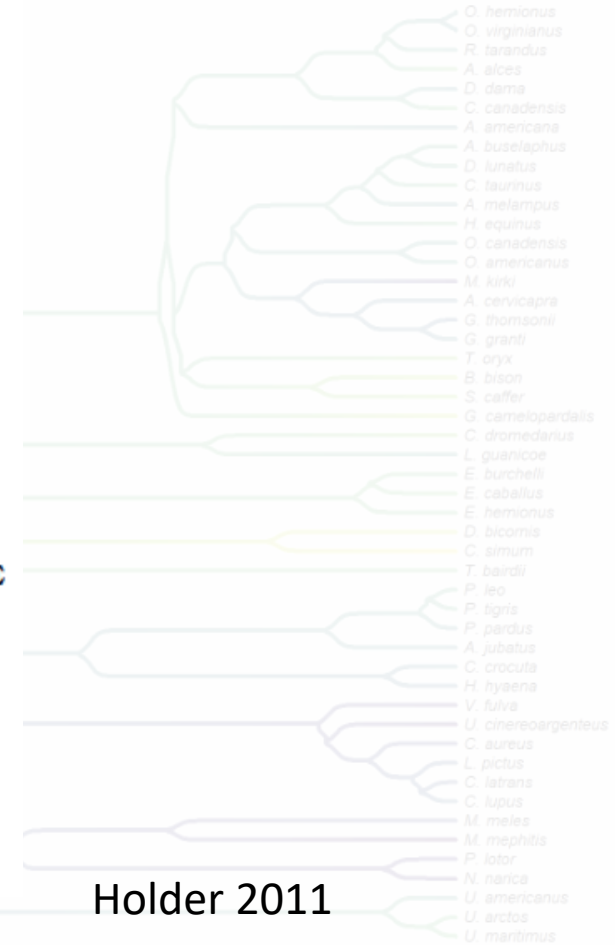


# Tree bisection reconnection (TBR)



TBR tries all possible rootings

ts, 743.7 kb)  
.53 4.54 1.48 1.95 ...



Holder 2011

```
1 library(phytools)
2 data(mammal.tree)
3 data(mammal.data)
4 head(mammal.data)
5 lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
6 obj<-contMap(mammal.tree,lnBodyMass,plot=FA
7 obj<-setMap(obj,viridisLite::viridis(n=8))
8 sigmoidPhylogram(obj$tree,colors=obj$cols,1
9
```

9.1 [Top Level] :

Console Terminal Background Jobs

R 4.2.2 - / -

R version 4.2.2 (2022-10-31 ucrt) -- "Innocent and Tru  
Copyright (C) 2022 The R Foundation for Statistical C  
Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANT  
You are welcome to redistribute it under certain cond  
Type 'license()' or 'licence()' for distribution deta

R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in pub

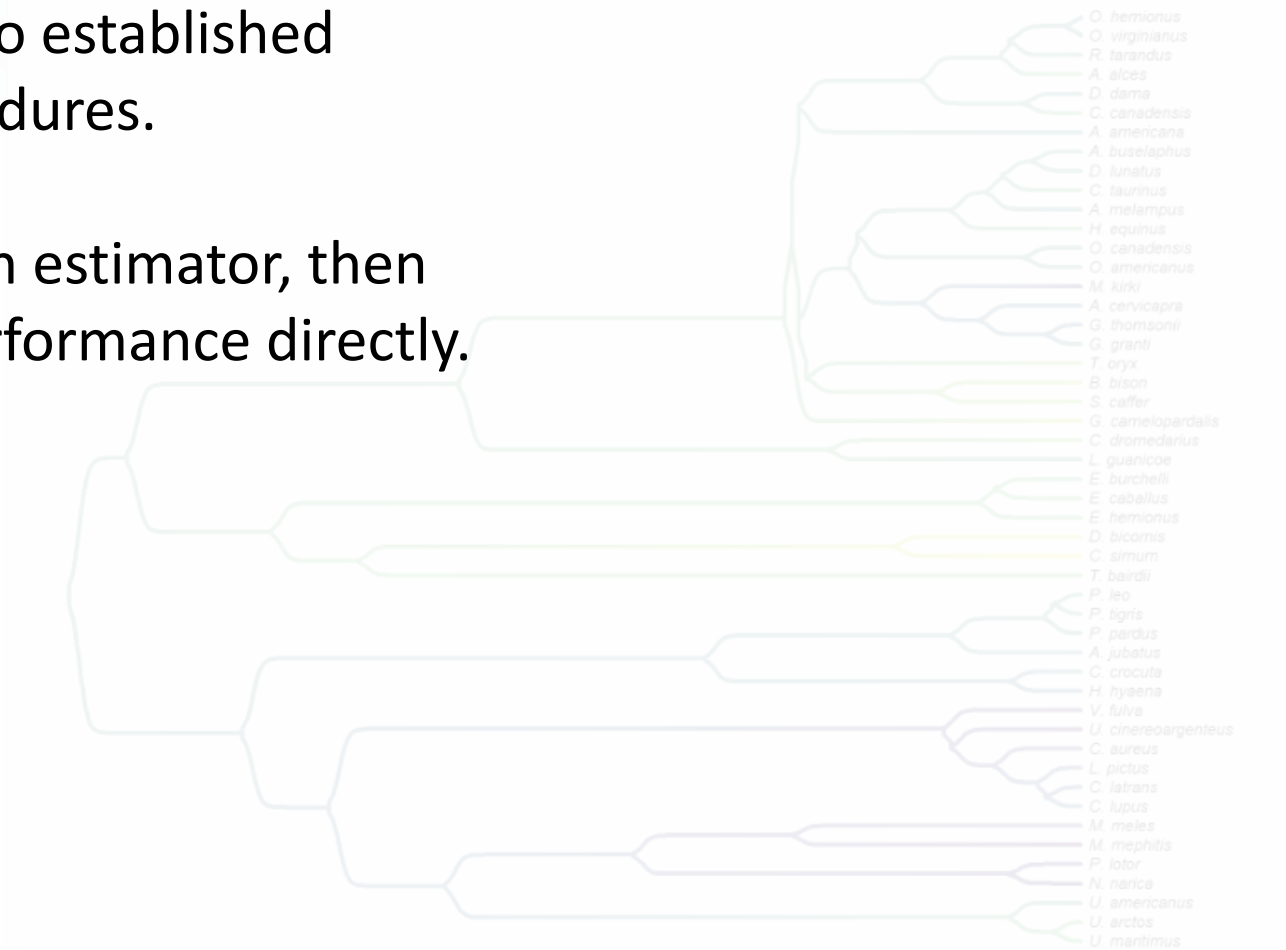
Type 'demo()' for some demos, 'help()' for on-line  
'help.start()' for an HTML browser interface to he  
Type 'q()' to quit R.

```
> library(phytools)
Loading required package: ape
Loading required package: maps
> data(mammal.tree)
> data(mammal.data)
> head(mammal.data)
      bodyMass homeRange
U. maritimus    265.0     115.60
U. arctos        251.3      82.80
U. americanus    93.4      56.80
N. narica         4.4       1.05
P. lotor          7.0       1.14
M. mephitis       2.5       2.50
> lnBodyMass<-setNames(log(mammal.data$bodyMass),rowni
> obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
> obj<-setMap(obj,viridisLite::viridis(n=8))
> sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftyj
> sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftype="i",fsize=0.7)
>
```

# Statistical properties of parsimony

## Assessing statistical properties –

- A) Via comparison to established estimation procedures.
- B) Treating MP as an estimator, then evaluating its performance directly.

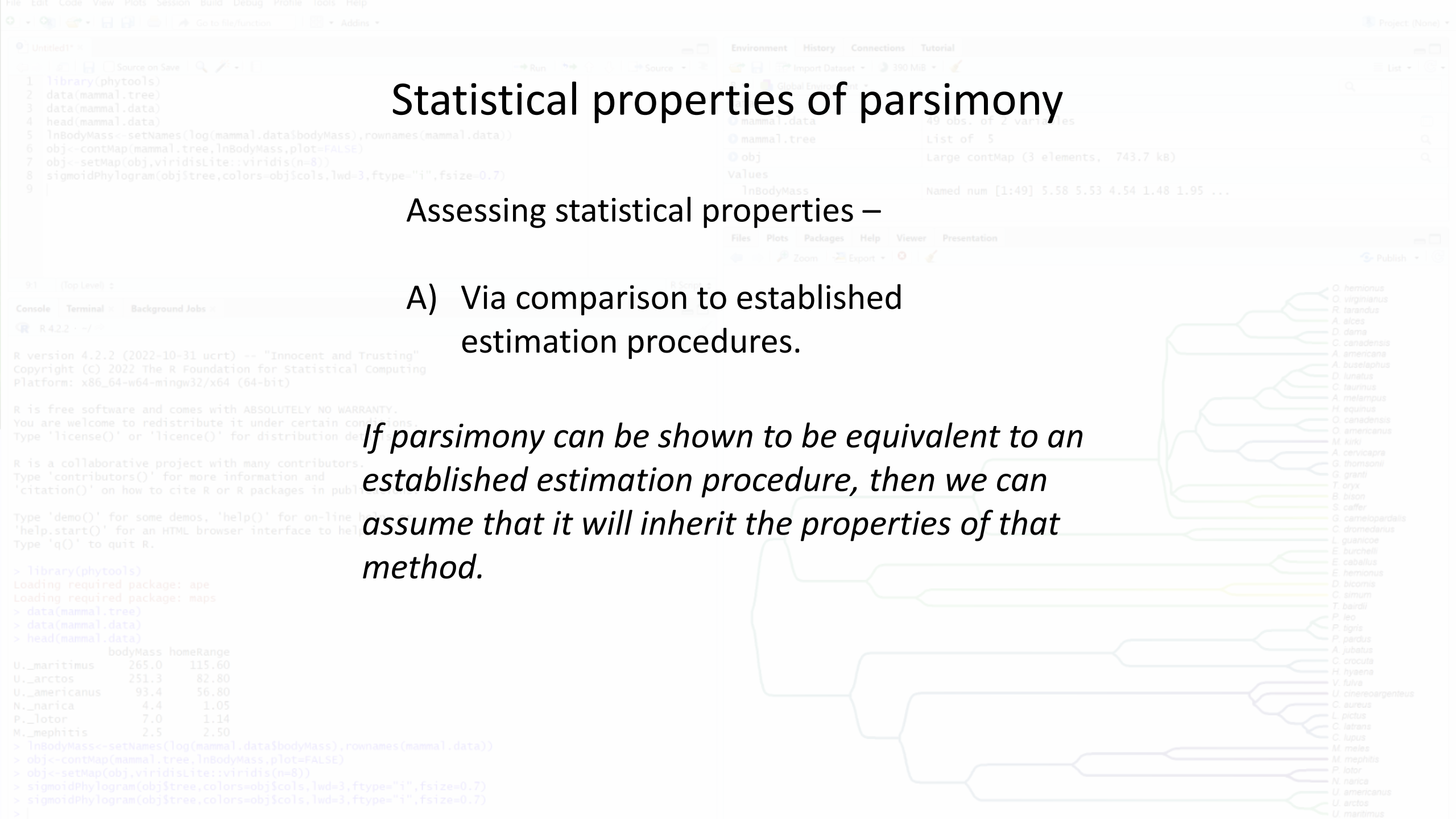


# Statistical properties of parsimony

Assessing statistical properties –

A) Via comparison to established estimation procedures.

*If parsimony can be shown to be equivalent to an established estimation procedure, then we can assume that it will inherit the properties of that method.*

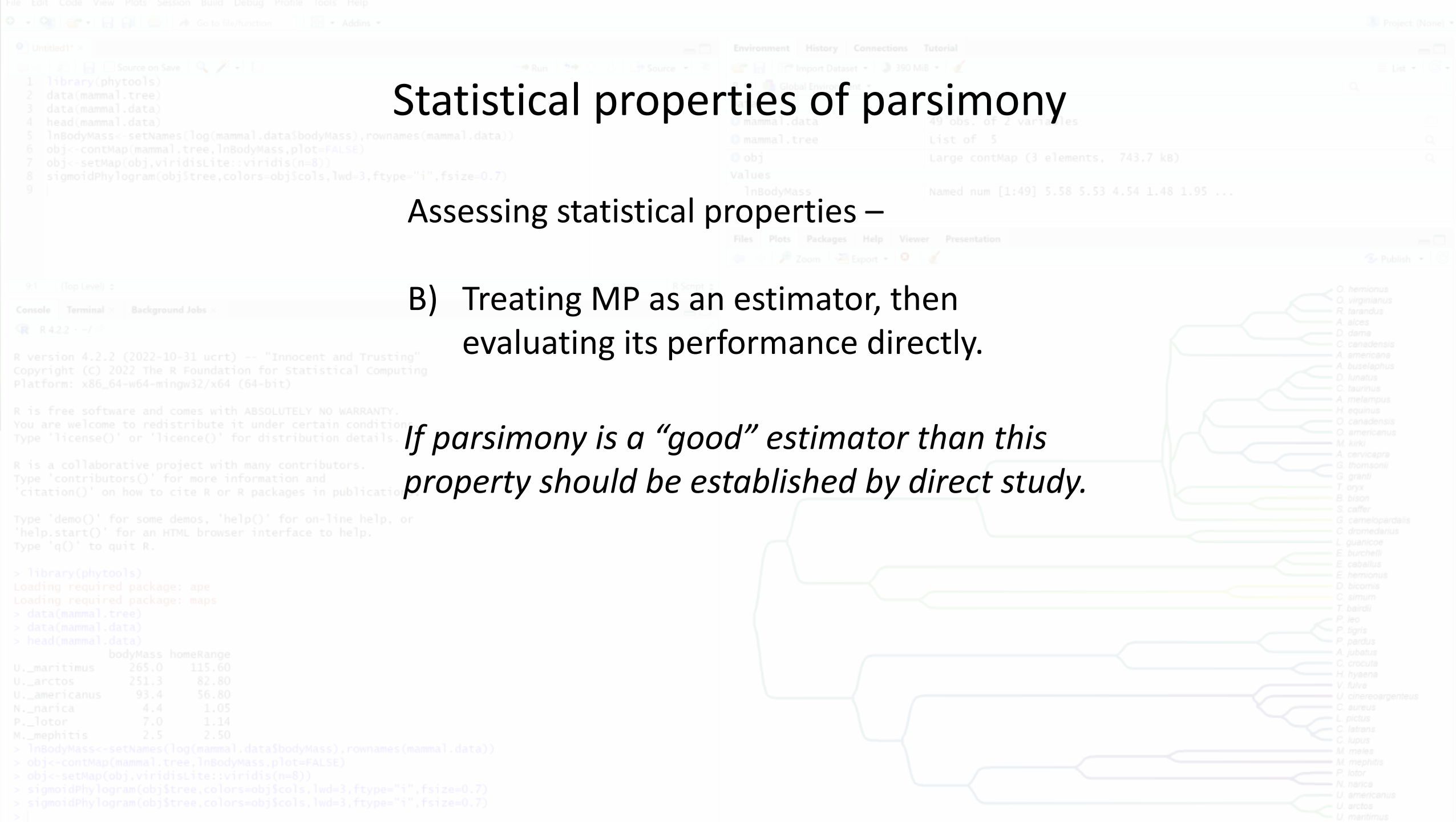


# Statistical properties of parsimony

Assessing statistical properties –

B) Treating MP as an estimator, then evaluating its performance directly.

*If parsimony is a “good” estimator then this property should be established by direct study.*





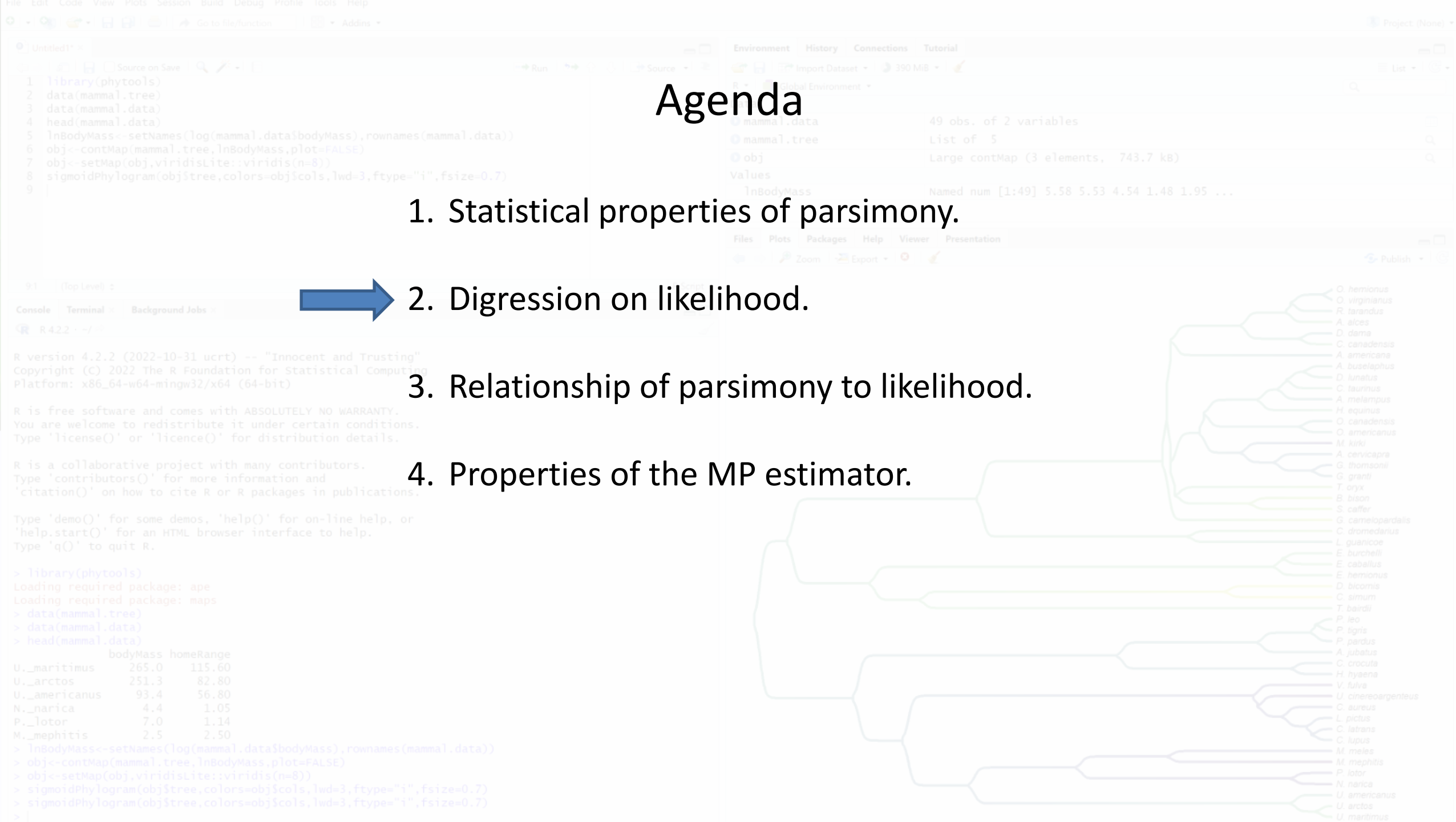
# Agenda

1. Statistical properties of parsimony.

➔ 2. Digression on likelihood.

3. Relationship of parsimony to likelihood.

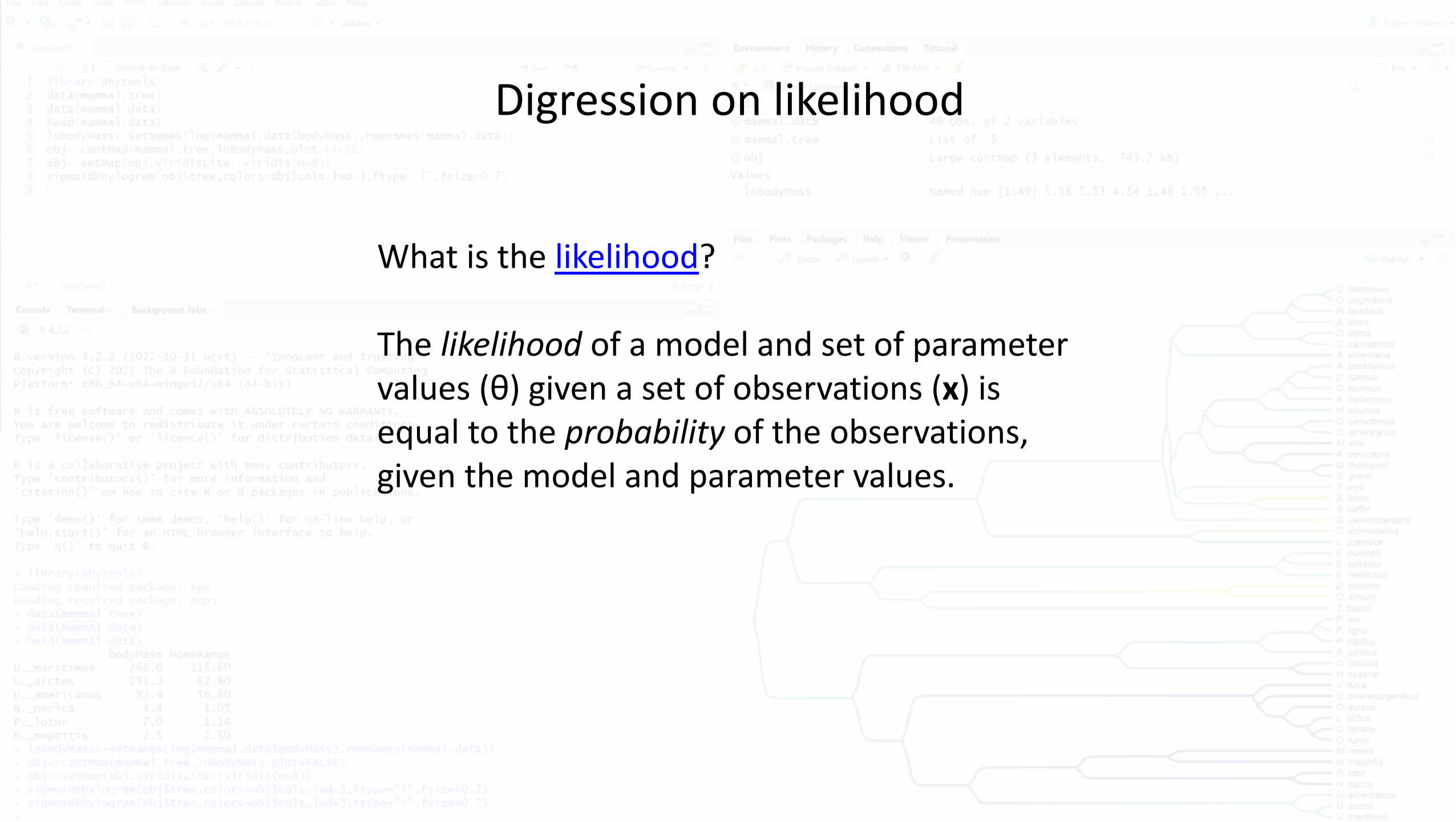
4. Properties of the MP estimator.



# Digression on likelihood

What is the [likelihood](#)?

The *likelihood* of a model and set of parameter values ( $\theta$ ) given a set of observations ( $\mathbf{x}$ ) is equal to the *probability* of the observations, given the model and parameter values.



# Digression on likelihood

What is the likelihood?

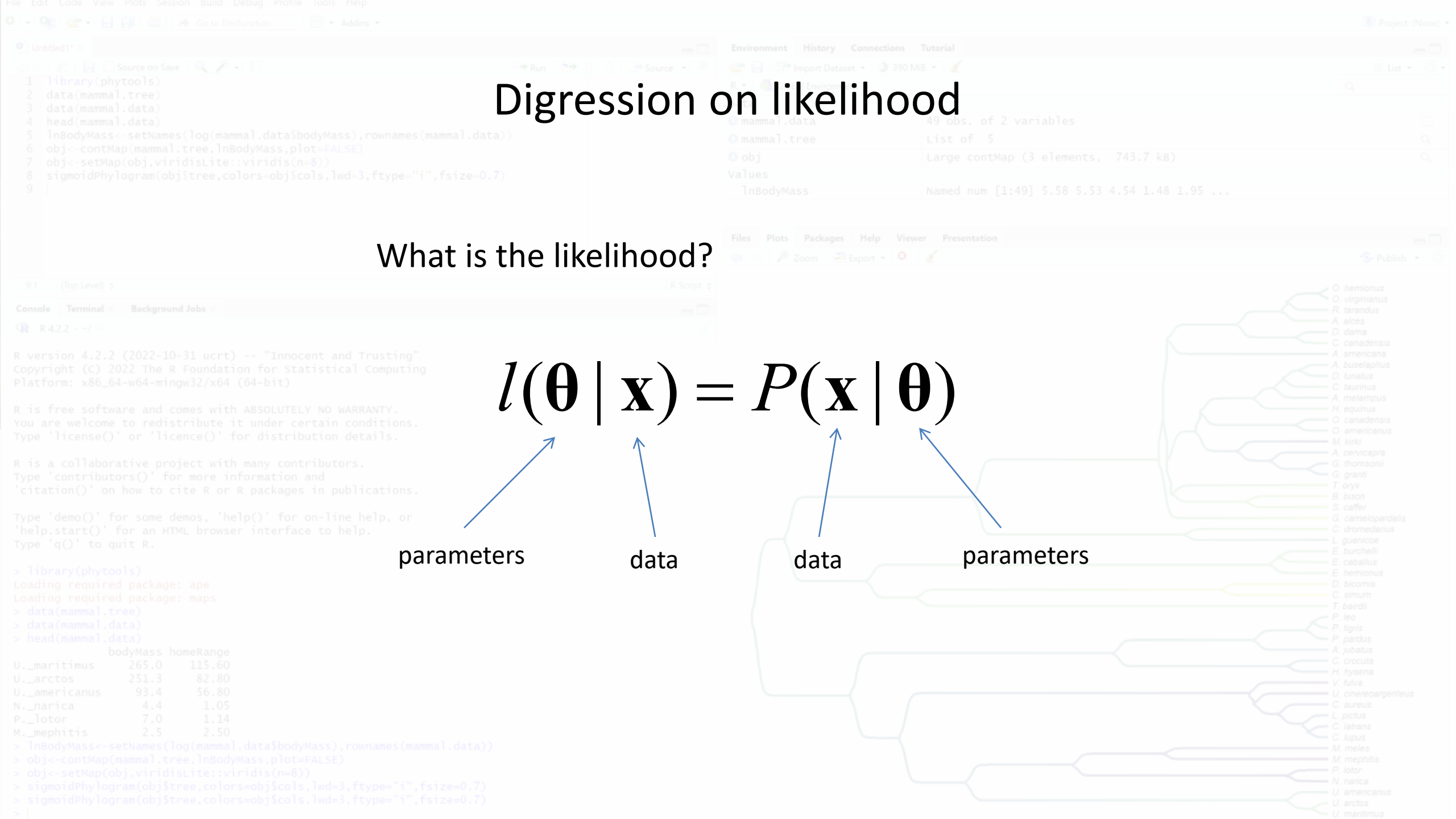
$$l(\boldsymbol{\theta} \mid \mathbf{x}) = P(\mathbf{x} \mid \boldsymbol{\theta})$$

parameters

data

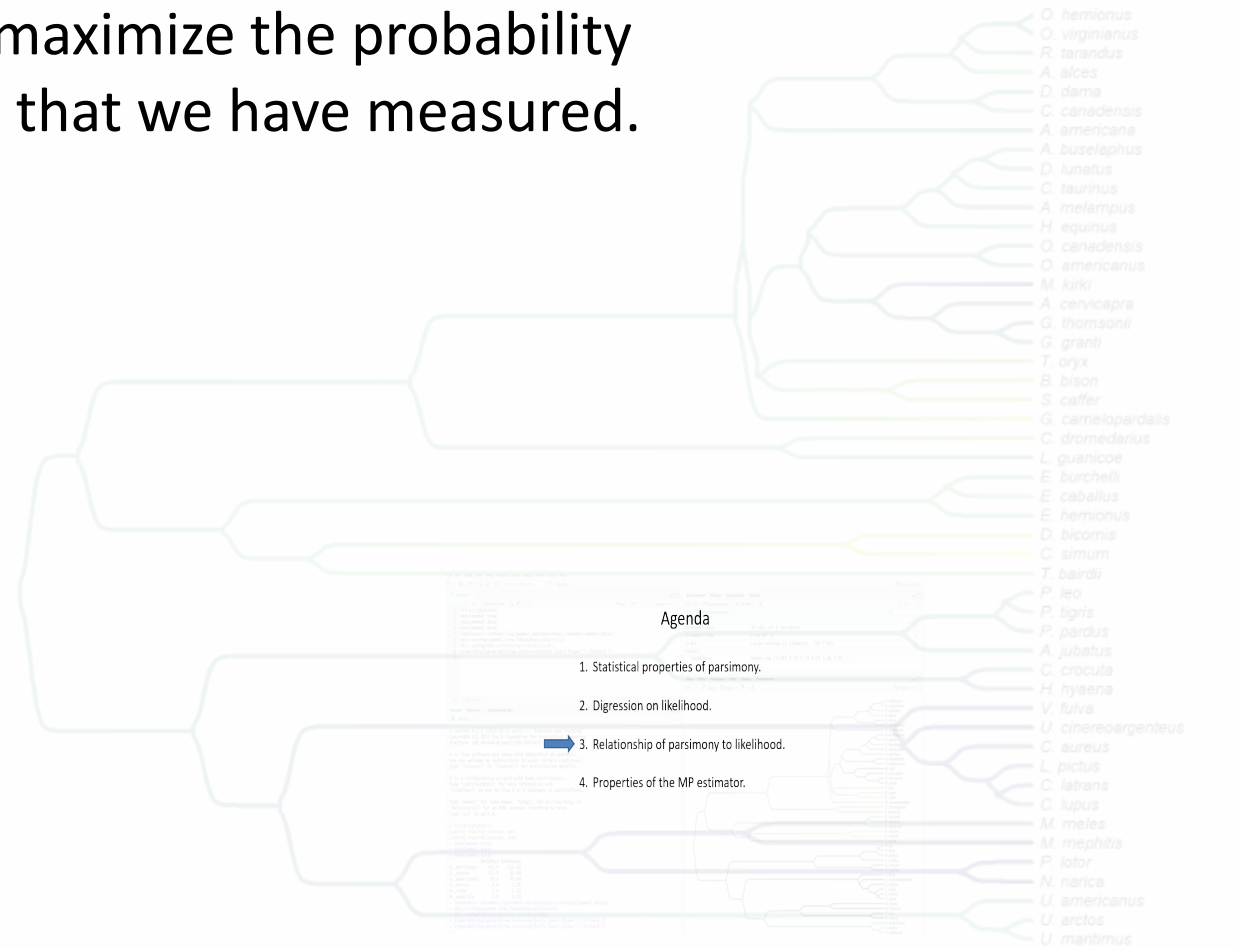
data

parameters



# Digression on likelihood

Maximum likelihood estimation involves identifying the model parameters that maximize the probability of having observed the data that we have measured.



# Digression on likelihood

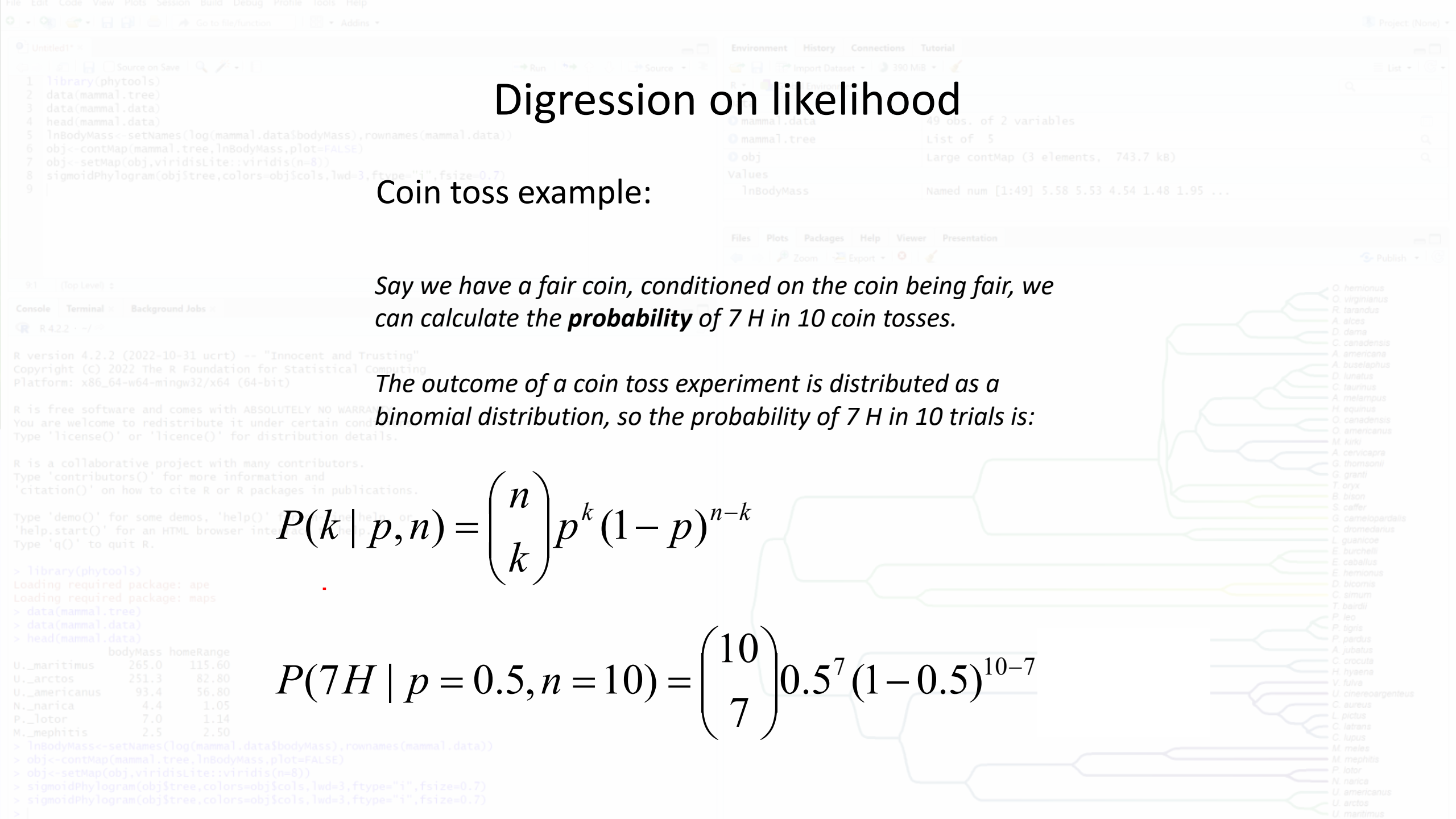
Coin toss example:

*Say we have a fair coin, conditioned on the coin being fair, we can calculate the **probability** of 7 H in 10 coin tosses.*

*The outcome of a coin toss experiment is distributed as a binomial distribution, so the probability of 7 H in 10 trials is:*

$$P(k | p, n) = \binom{n}{k} p^k (1-p)^{n-k}$$

$$P(7H | p = 0.5, n = 10) = \binom{10}{7} 0.5^7 (1-0.5)^{10-7}$$



# Digression on likelihood

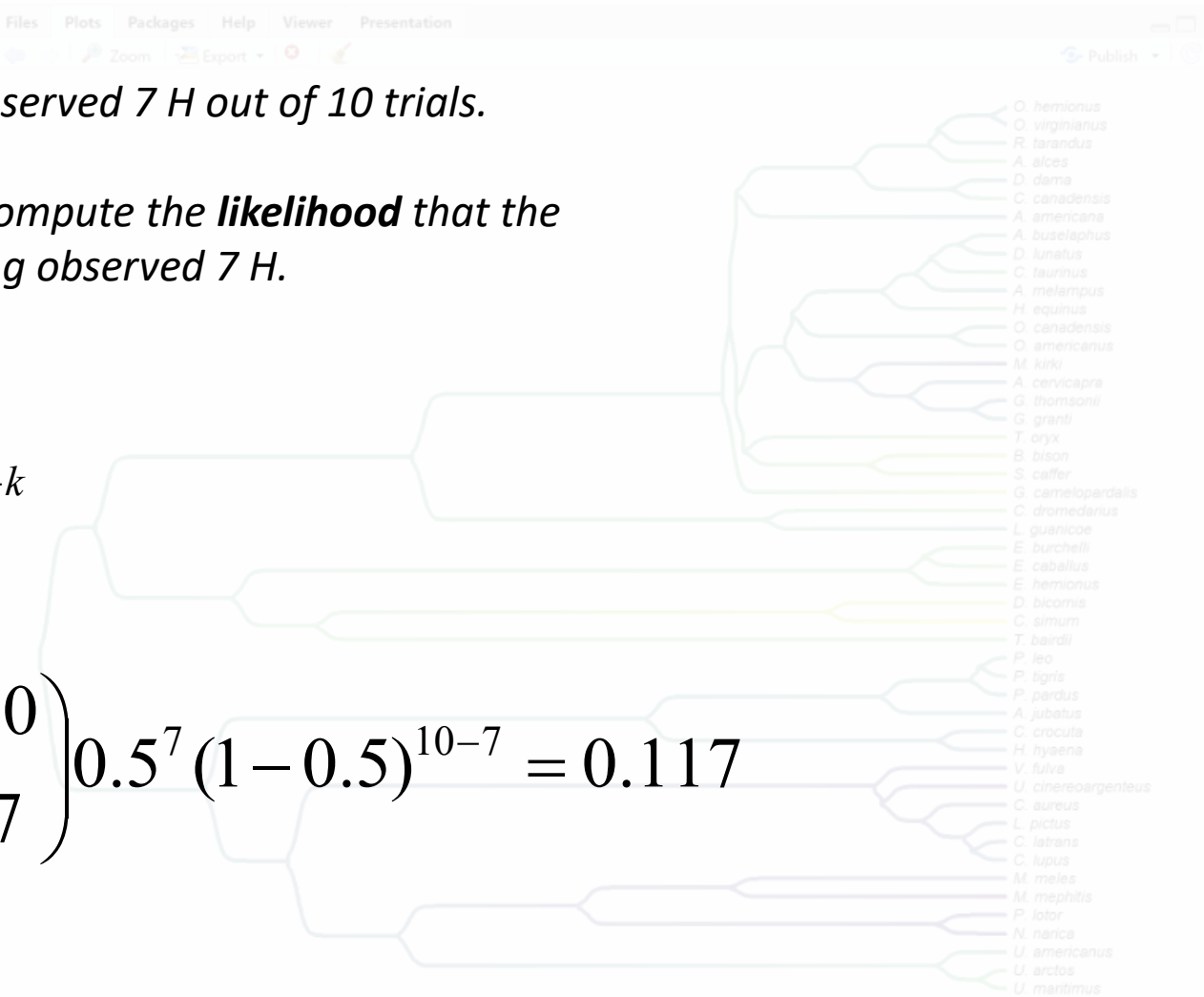
Coin toss example:

*Now conversely, say we have observed 7 H out of 10 trials.*

*In this case, we might want to compute the **likelihood** that the coin is fair, conditioned on having observed 7 H.*

$$l(p | k, n) = \binom{n}{k} p^k (1-p)^{n-k}$$

$$l(p = 0.5 | 7H, n = 10) = \binom{10}{7} 0.5^7 (1-0.5)^{10-7} = 0.117$$

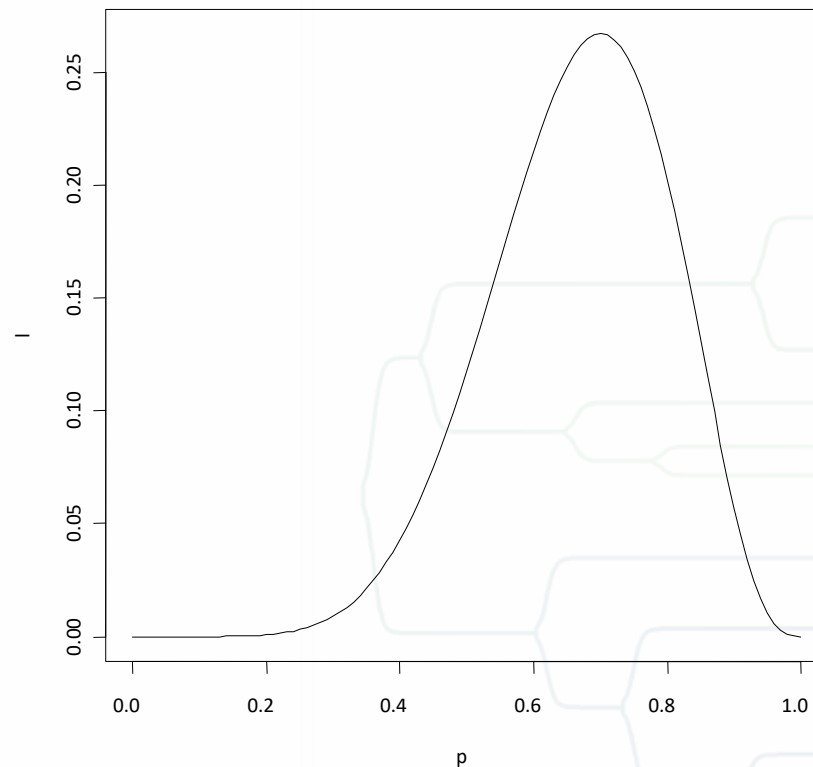




# Digression on likelihood

Coin toss example:

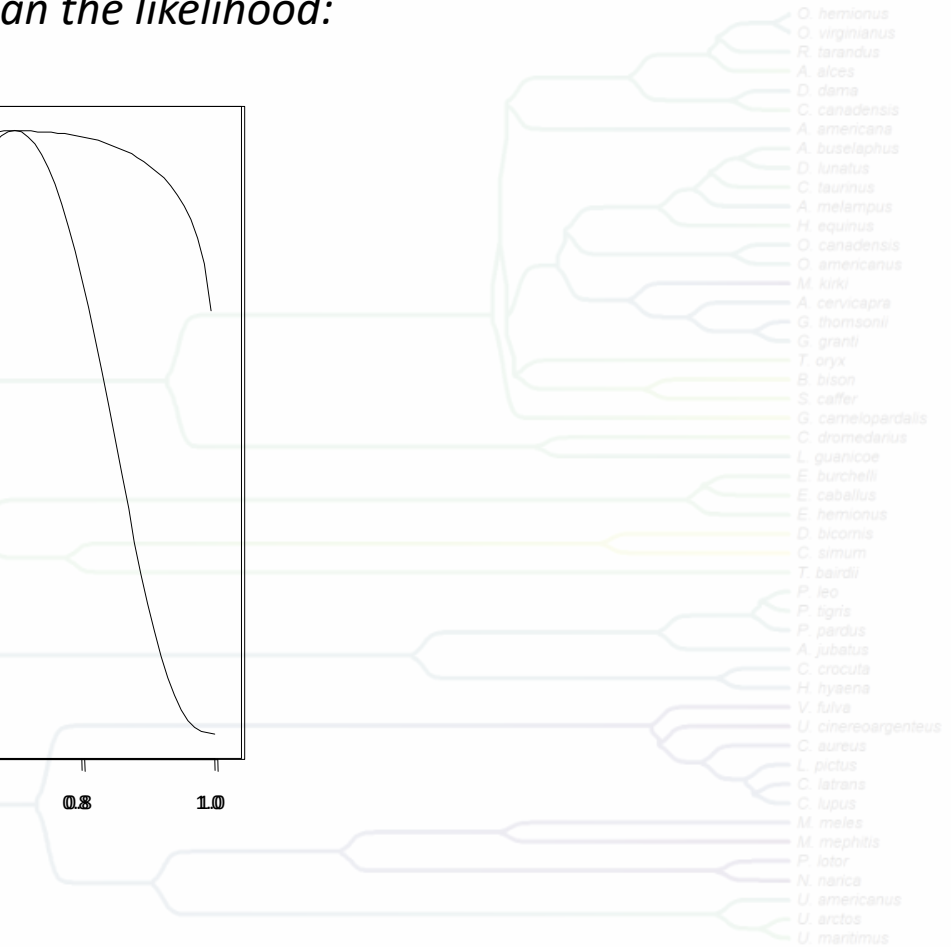
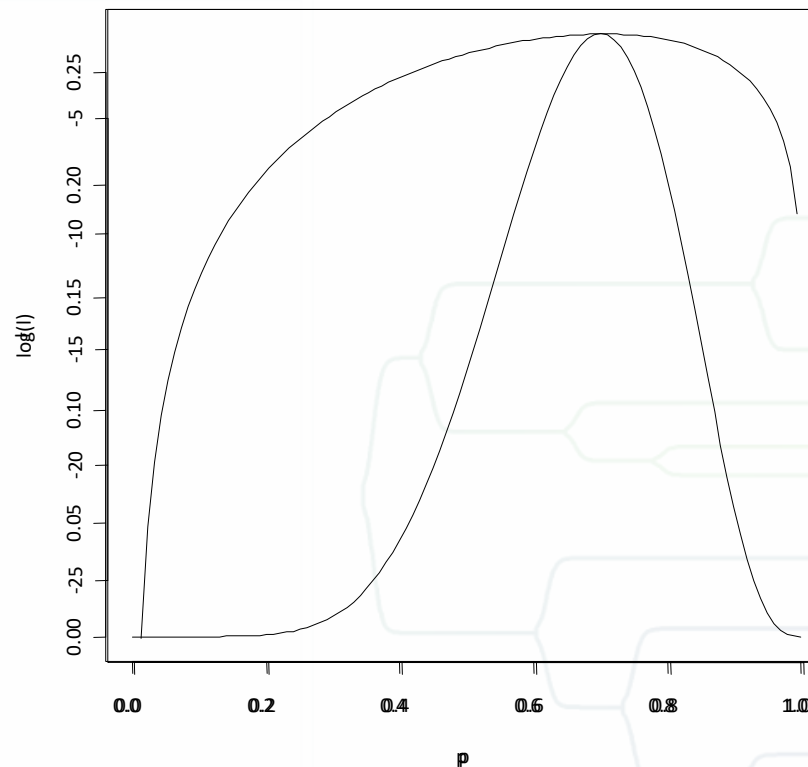
Given  $7H$  we can **plot** the likelihood across the possible range of  $p$ , i.e.:



# Digression on likelihood

Coin toss example:

Typically, we plot the ***log(likelihood)*** rather than the likelihood:

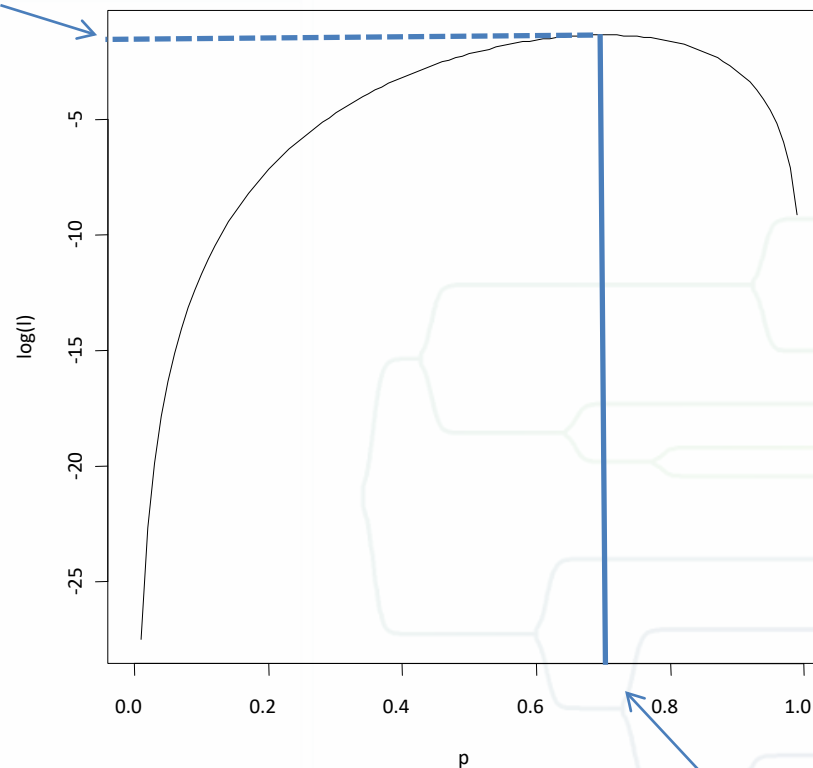


# Digression on likelihood

Coin toss example:

The **maximum likelihood estimate (MLE)** is the value of  $p$  that maximizes the likelihood of our model:

-1.32



MLE( $p$ ) = 0.7

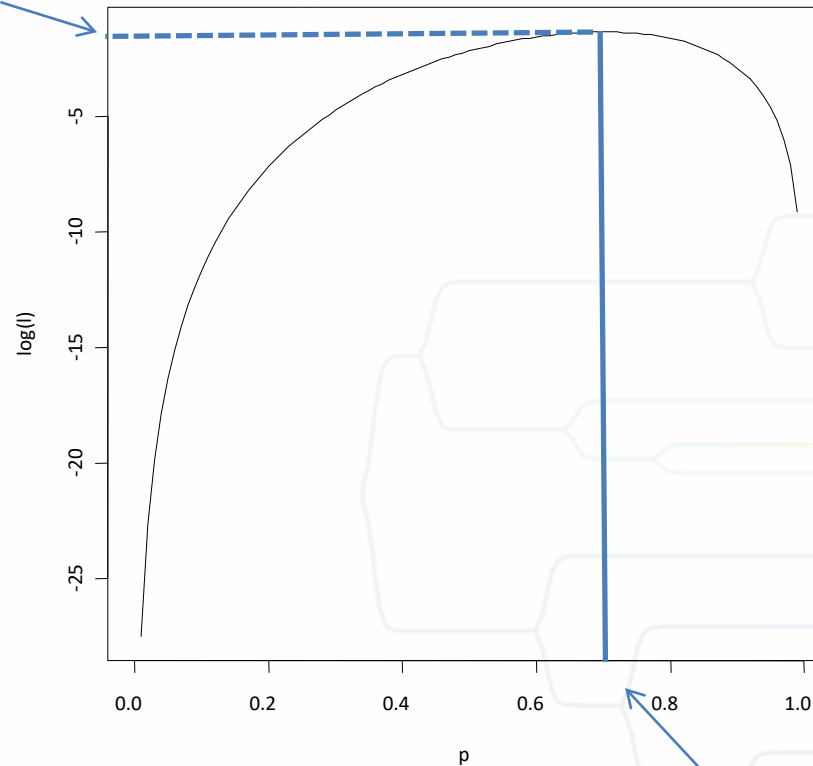
Phylogenetic tree showing relationships between various species, including *O. hemionus*, *O. virginianus*, *R. tarandus*, *A. alces*, *D. dama*, *C. canadensis*, *A. americana*, *A. buselaphus*, *D. lunatus*, *C. taurinus*, *A. melampus*, *H. equinus*, *O. canadensis*, *O. americanus*, *M. kirkii*, *A. cervicapra*, *G. thomsonii*, *G. granti*, *T. oryx*, *B. bison*, *S. caffer*, *G. camelopardalis*, *O. dromedarius*, *L. guanicoe*, *E. burchelli*, *E. caballus*, *E. hemionus*, *D. bicornis*, *C. simum*, *T. bairdi*, *P. leo*, *P. tigris*, *P. pardus*, *A. jubatus*, *C. crocuta*, *H. hyaena*, *V. fulva*, *U. cinereoargenteus*, *C. aureus*, *L. pictus*, *C. latrans*, *C. lupus*, *M. meles*, *M. mephitis*, *P. lotor*, *N. narica*, *U. americanus*, *U. arctos*, and *U. maritimus*.

# Digression on likelihood

## Coin toss example:

The MLEs tends to have “good” statistical properties, for instance: **normality, efficiency, consistency, and minimum variance.**

-1.32



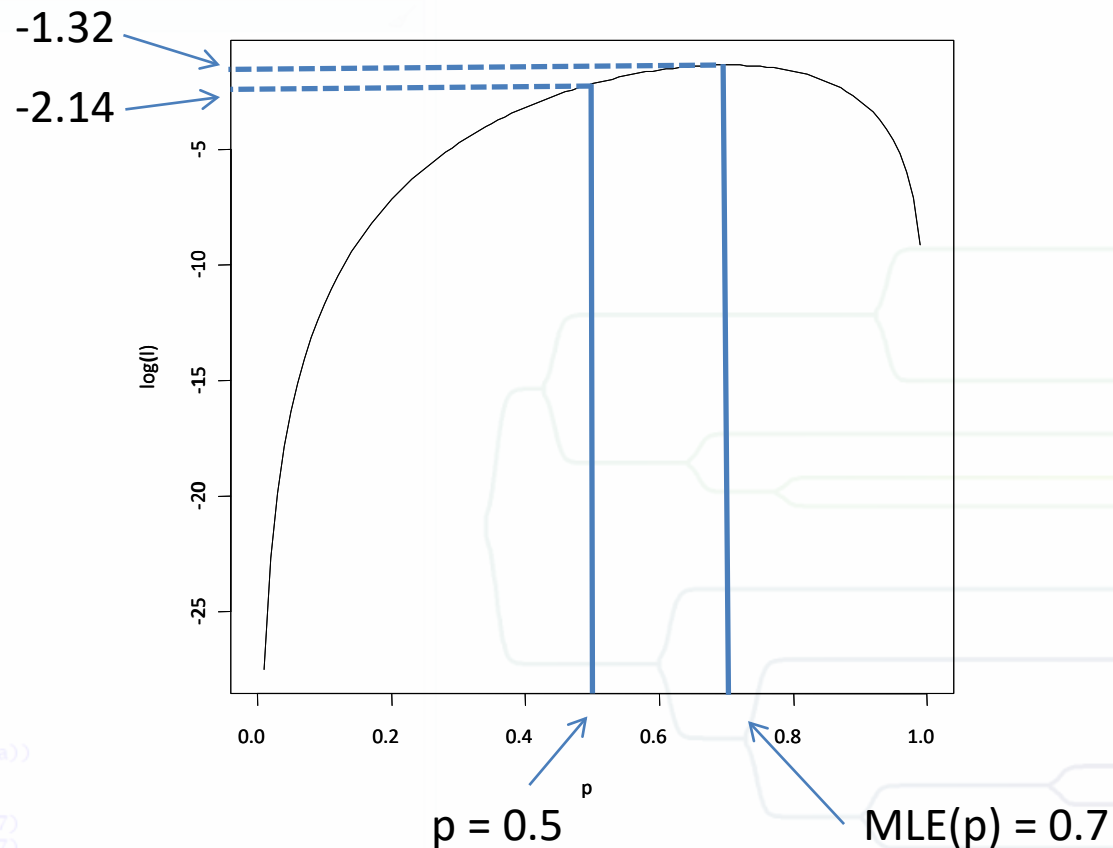
MLE(p) = 0.7



# Digression on likelihood

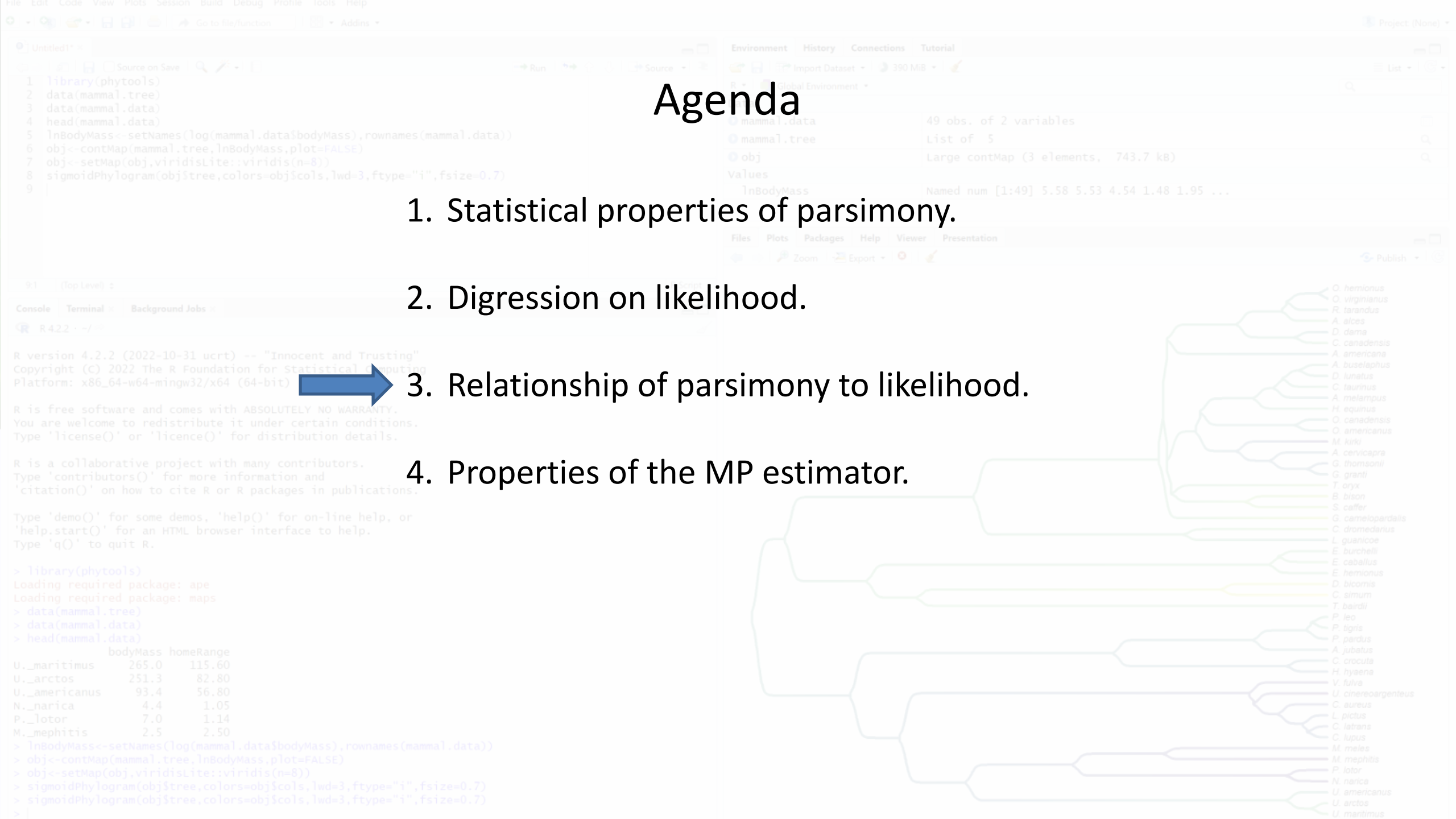
Coin toss example:

*Log-likelihoods can also be compared among models, for instance:*



# Agenda

1. Statistical properties of parsimony.
2. Digression on likelihood.
3. Relationship of parsimony to likelihood.
4. Properties of the MP estimator.





# Punchline

- We can *prove* that MP will yield the MLE of the tree, when the rates of character change (evolution) is very slow.
- MP has also been shown (more generally) to be an MLE of the tree under a model called the ‘no common mechanism’ model; however this model has too many parameters to inherit the good statistical properties of ML.

## Agenda

1. Statistical properties of parsimony.
2. Digression on likelihood.
3. Relationship of parsimony to likelihood.
- ➡ 4. Properties of the MP estimator.

# Relationship between parsimony & ML

To compute the likelihood we need a model, so we will use a random process called a [Markov process](#) as the basis for our model.

Under this process, the probability that a change in character  $i$  will result along branch  $j$  is given by:

$$P(1 | 0, t_j) = \frac{1}{2} \left( 1 - e^{-2r_i t_j} \right)$$

For low enough  $r_i$  this is nearly equal to:

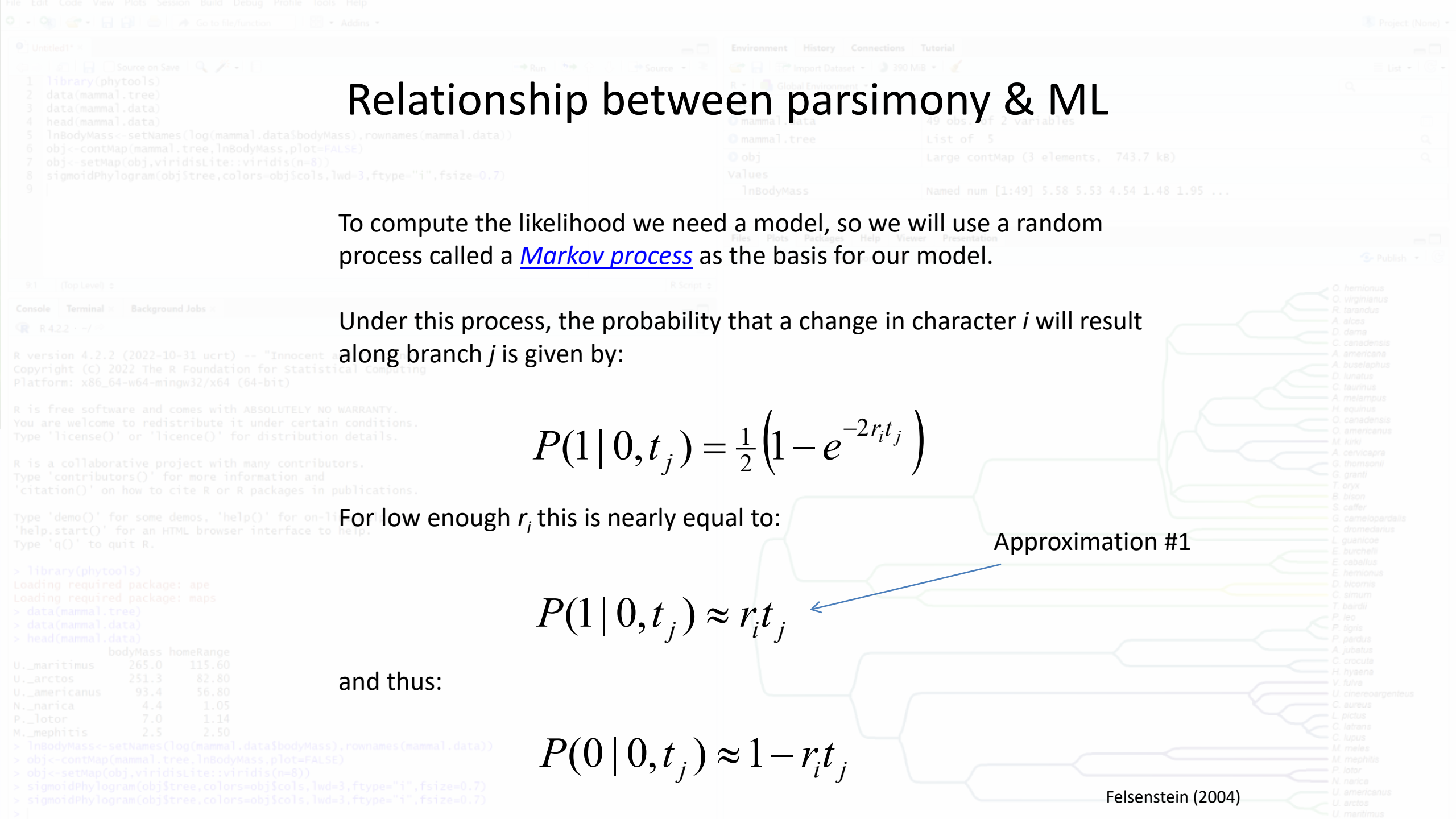
$$P(1 | 0, t_j) \approx r_i t_j$$

and thus:

$$P(0 | 0, t_j) \approx 1 - r_i t_j$$

Approximation #1

Felsenstein (2004)

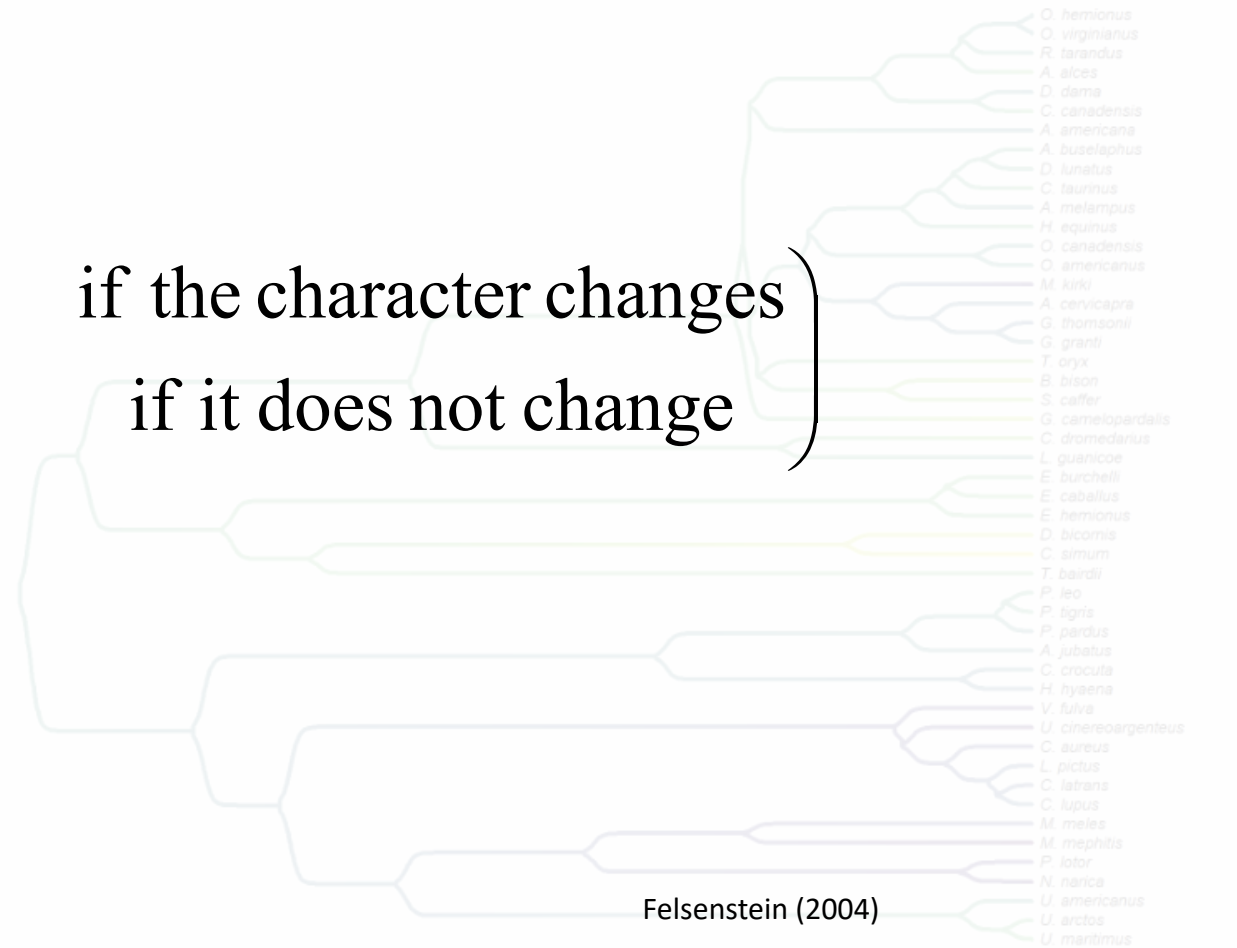


# Relationship between parsimony & ML

We can compute the likelihood as the probability of our data given the model and parameters (in this case, our tree):

$$l = P(Data | Tree)$$

$$= \prod_{i=1}^{chars} \sum_{recon} \left( \frac{1}{2} \prod_{j=1}^{branches} \begin{cases} r_i t_j & \text{if the character changes} \\ 1 - r_i t_j & \text{if it does not change} \end{cases} \right)$$



Felsenstein (2004)

# Relationship between parsimony & ML

If we assume, as we have already done, that  $r_i$  is very small, then we can effectively assume the following:

$$l = \prod_{i=1}^{chars} \sum_{recon} \left( \frac{1}{2} \prod_{j=1}^{branches} \begin{cases} r_i t_j & \text{if the character changes} \\ 1 - r_i t_j & \text{if it does not change} \end{cases} \right)$$

and then:

$$l = \prod_{i=1}^{chars} \sum_{recon} \left( \frac{1}{2} \prod_{j=1}^{branches} (r_i t_j)^{n_{ij}} \right)$$

where  $n_{ij}$  is merely the *number* of changes in the  $i$ th character on the  $j$ th branch for a given reconstruction (either 0 or 1 in this case).

Approximation #2

# Relationship between parsimony & ML

Next, if we assume that the product of the  $r_i t_j$ s differs widely among different reconstructions, such that in most reconstructions:

$$\prod_{j=1}^{\text{branches}} (r_i t_j)^{n_{ij}} \approx 0 \quad \leftarrow \text{Approximation \#3}$$

then we can say that:

$$l \approx \prod_{i=1}^{\text{chars}} \prod_{j=1}^{\text{branches}} (r_i t_j)^{n_{ij}}$$

for only the reconstruction with the *smallest* number of changes (dropping the factor of  $\frac{1}{2}$  which will be the same across all reconstructions).

Felsenstein (2004)

# Relationship between parsimony & ML

If we take the negative logarithm of both sides, we get:

$$-L \approx \sum_{i=1}^{chars} \sum_{j=1}^{branches} n_{ij} [-\ln(r_i t_j)]$$

where  $L$  is the log-likelihood.

This is just the parsimony score, times a set of weights given by  $-\ln(r_i t_j)$ .

Thus, we have proved that **weighted parsimony is a likelihood method** under several assumptions.

Felsenstein (2004)



# Relationship between parsimony & ML

Finally, as:

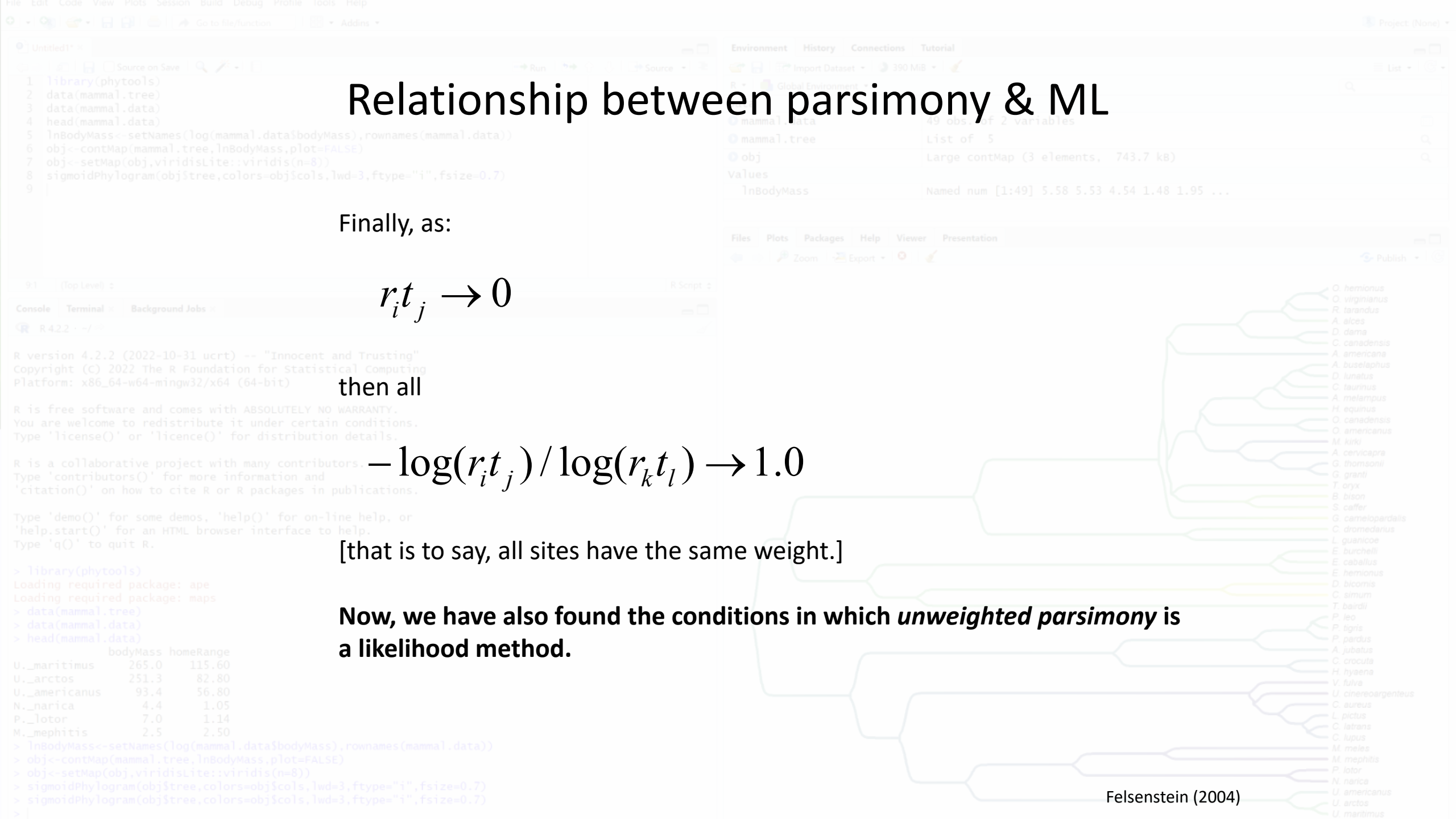
$$r_i t_j \rightarrow 0$$

then all

$$-\log(r_i t_j) / \log(r_k t_l) \rightarrow 1.0$$

[that is to say, all sites have the same weight.]

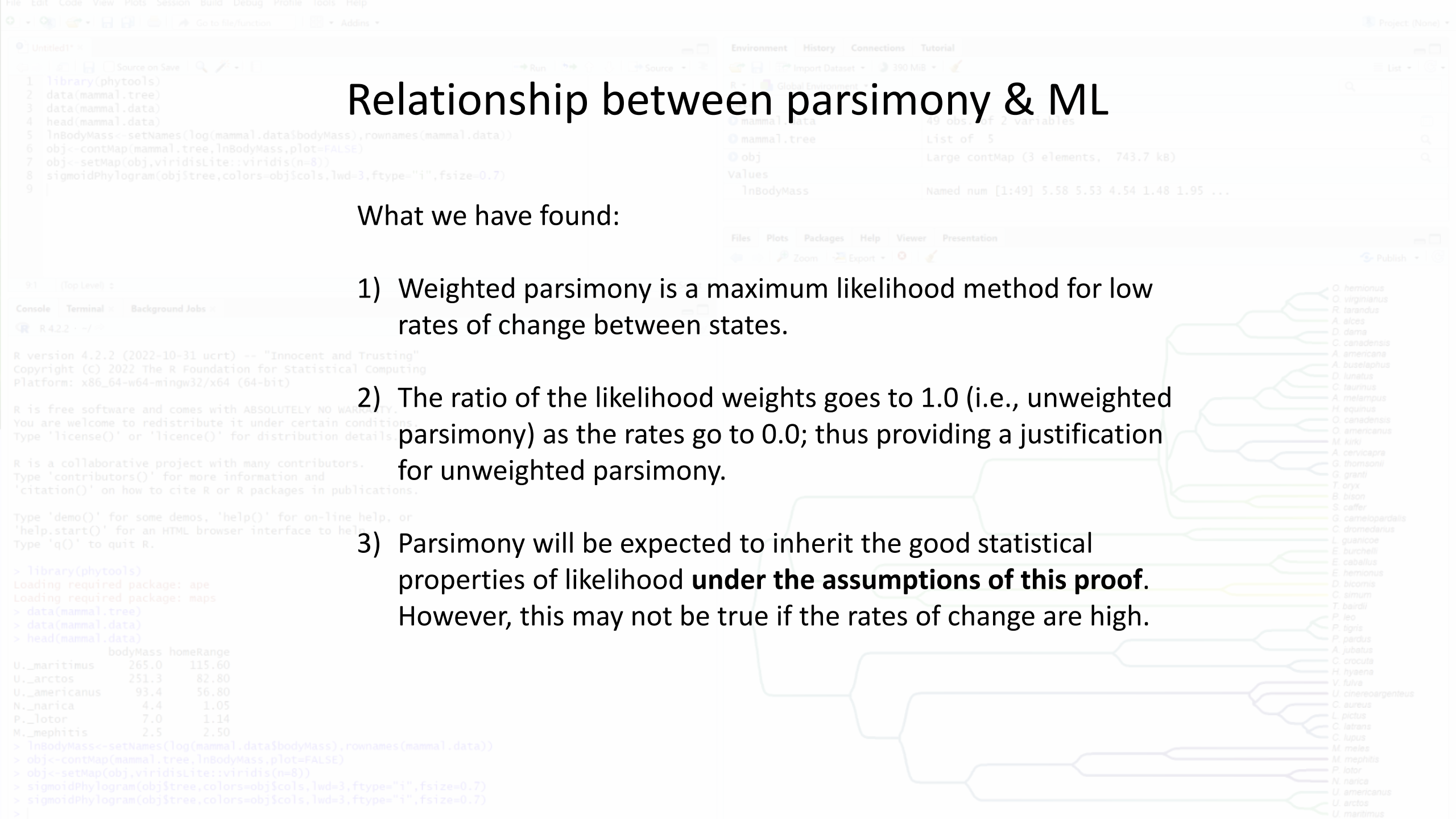
Now, we have also found the conditions in which *unweighted parsimony* is a likelihood method.



# Relationship between parsimony & ML

What we have found:

- 1) Weighted parsimony is a maximum likelihood method for low rates of change between states.
- 2) The ratio of the likelihood weights goes to 1.0 (i.e., unweighted parsimony) as the rates go to 0.0; thus providing a justification for unweighted parsimony.
- 3) Parsimony will be expected to inherit the good statistical properties of likelihood **under the assumptions of this proof**. However, this may not be true if the rates of change are high.



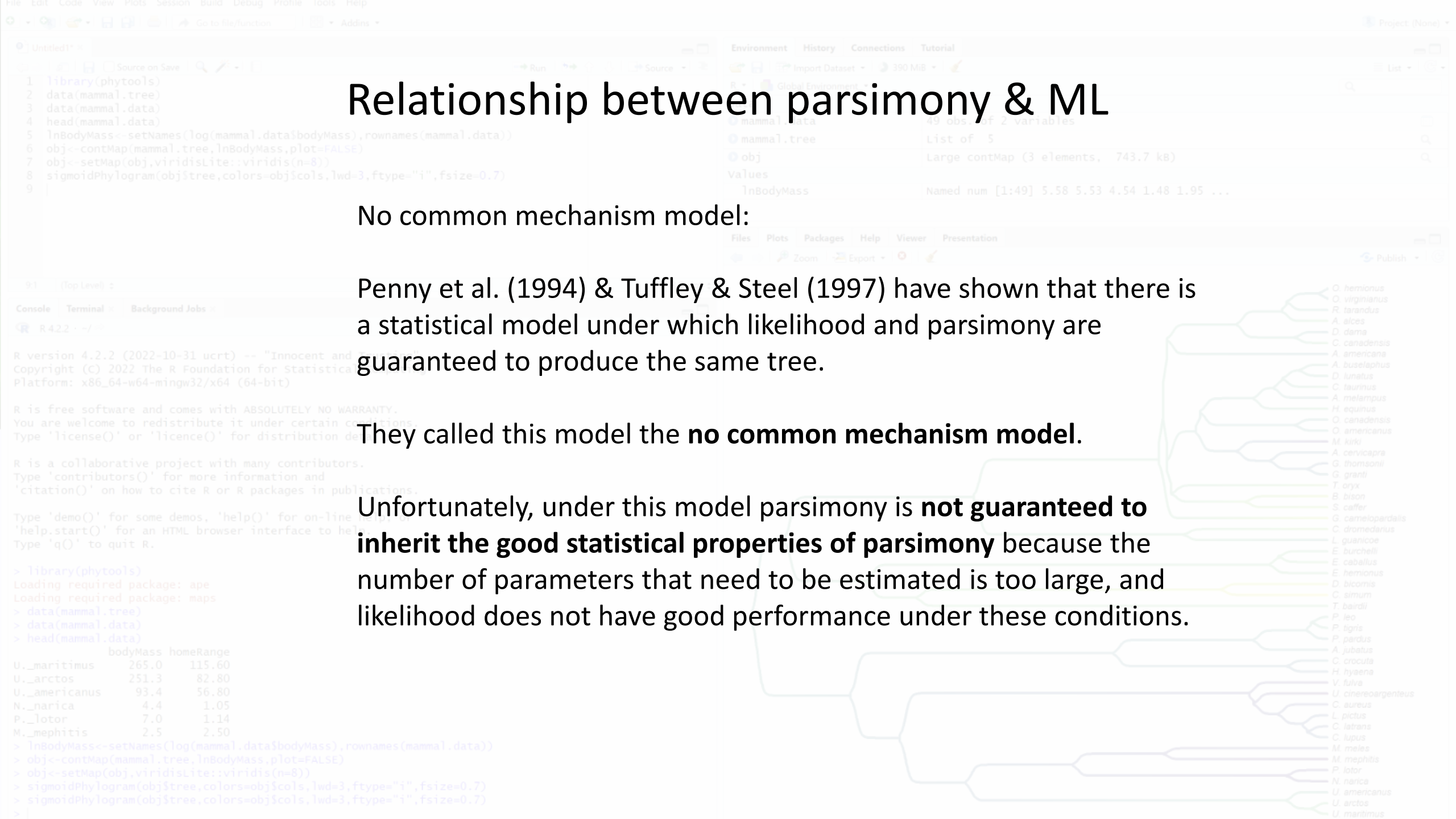
# Relationship between parsimony & ML

No common mechanism model:

Penny et al. (1994) & Tuffley & Steel (1997) have shown that there is a statistical model under which likelihood and parsimony are guaranteed to produce the same tree.

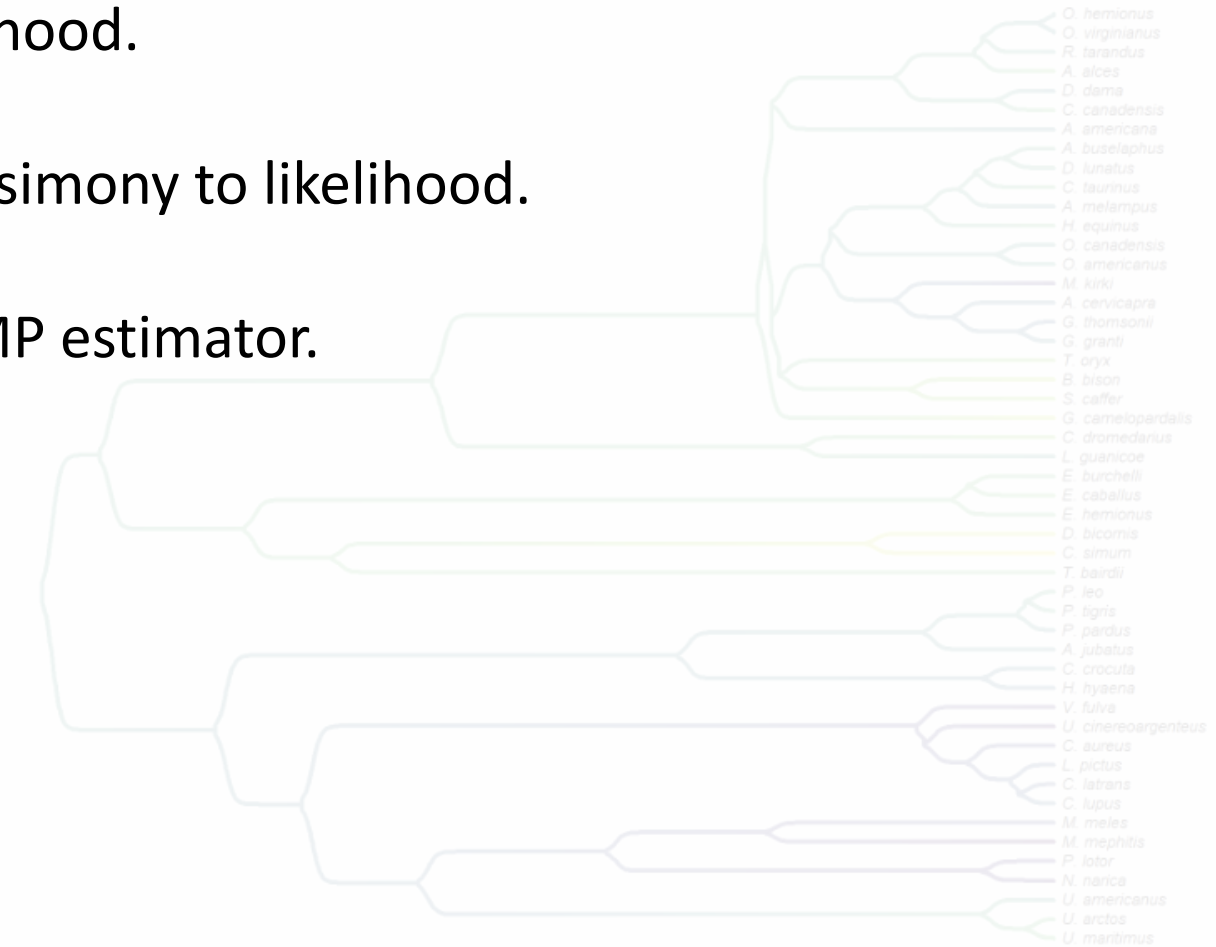
They called this model the **no common mechanism model**.

Unfortunately, under this model parsimony is **not guaranteed to inherit the good statistical properties of parsimony** because the number of parameters that need to be estimated is too large, and likelihood does not have good performance under these conditions.



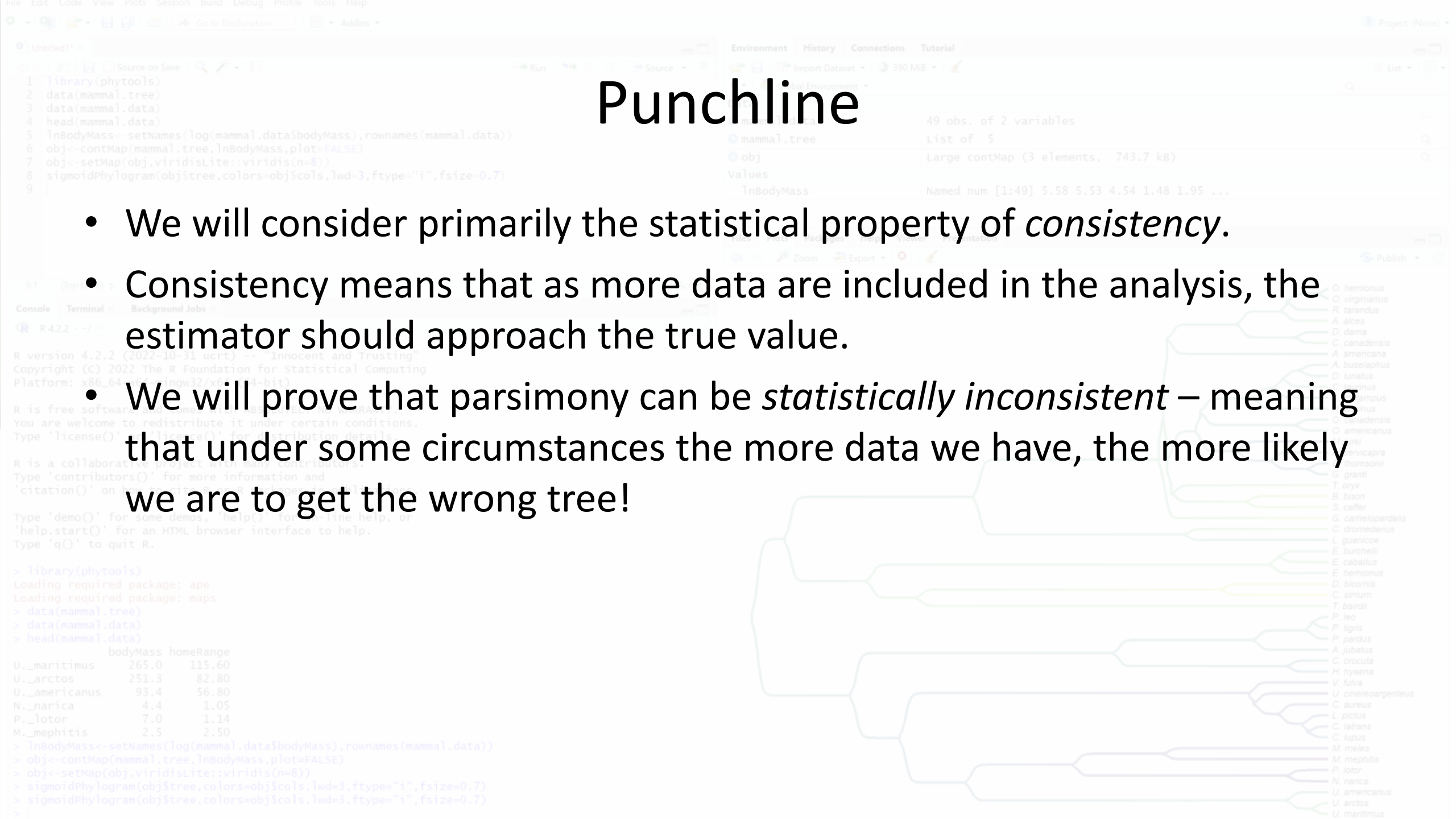
# Agenda

1. Statistical properties of parsimony.
2. Digression on likelihood.
3. Relationship of parsimony to likelihood.
4. Properties of the MP estimator.



# Punchline

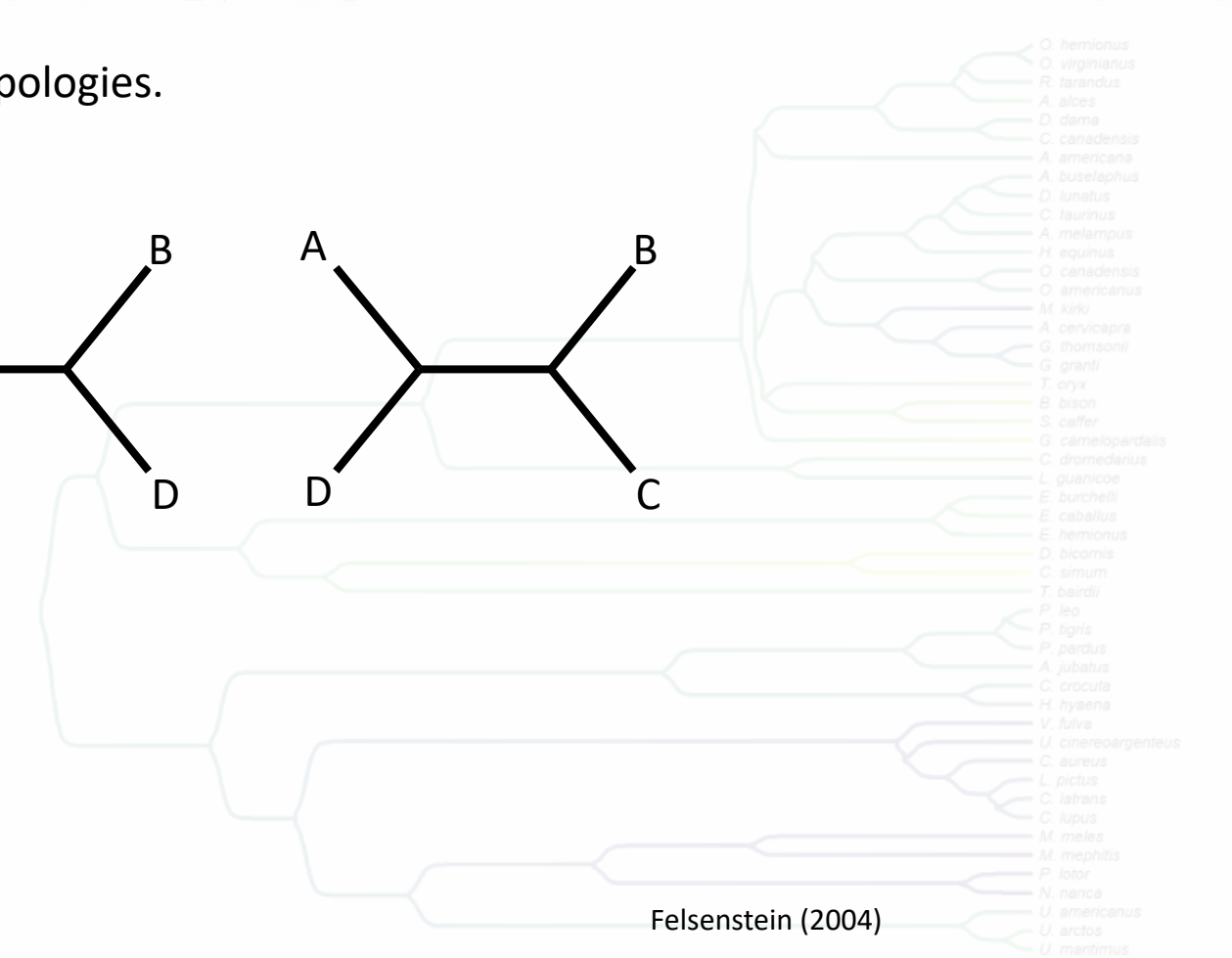
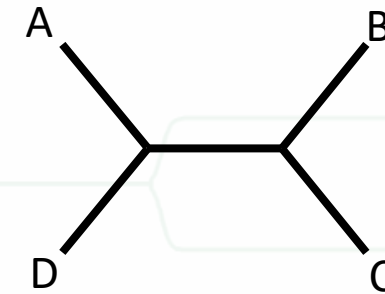
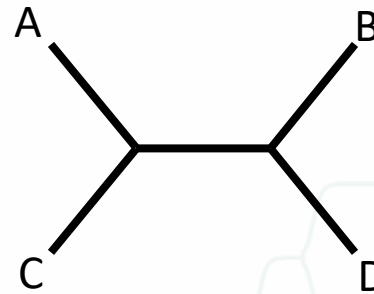
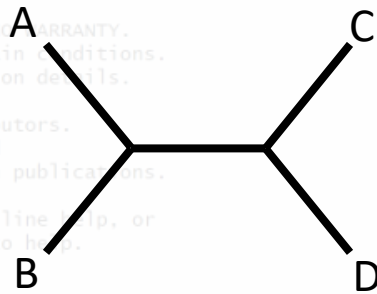
- We will consider primarily the statistical property of *consistency*.
- Consistency means that as more data are included in the analysis, the estimator should approach the true value.
- We will prove that parsimony can be *statistically inconsistent* – meaning that under some circumstances the more data we have, the more likely we are to get the wrong tree!



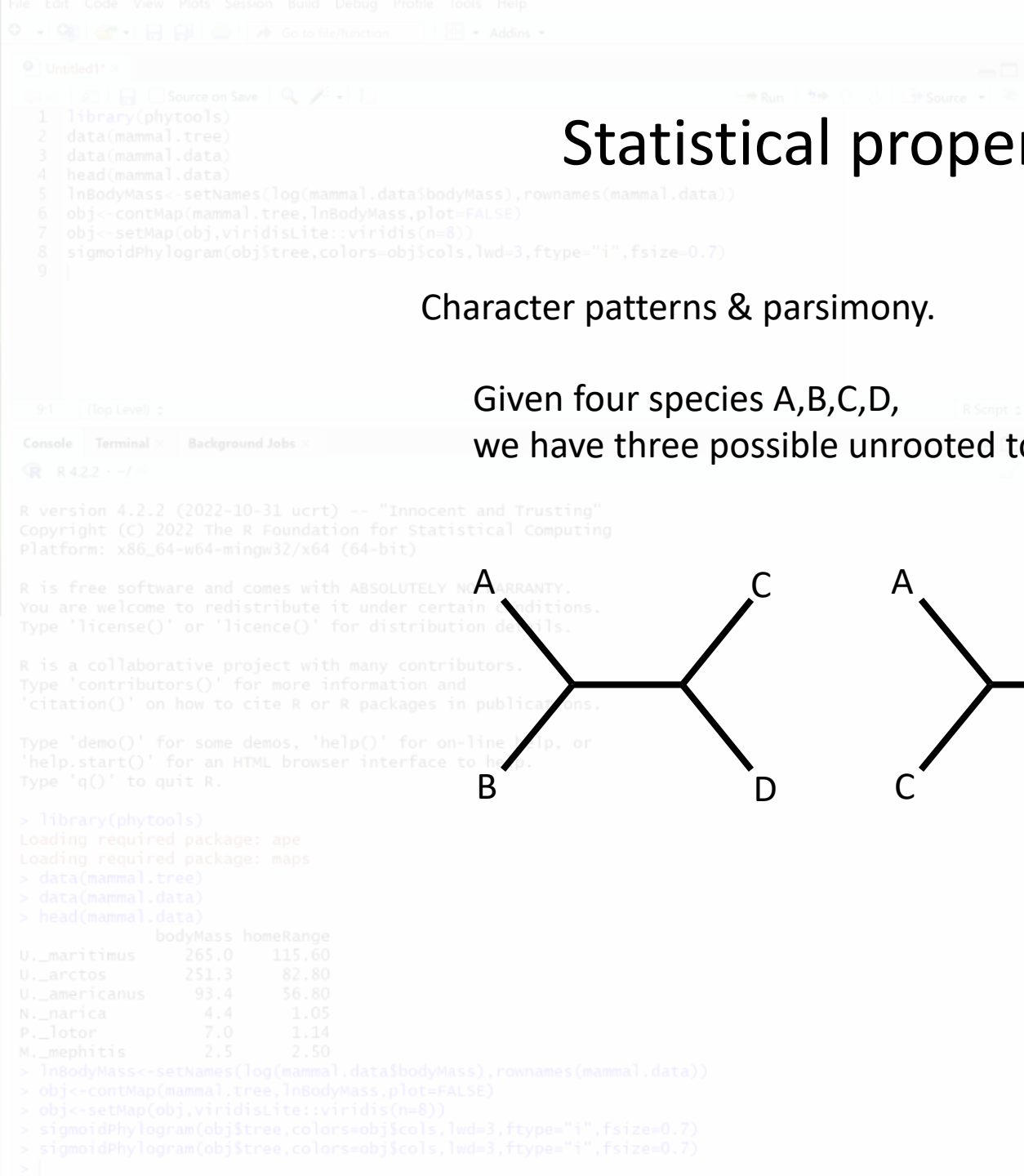
# Statistical properties of parsimony

Character patterns & parsimony.

Given four species A,B,C,D,  
we have three possible unrooted topologies.



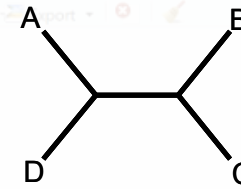
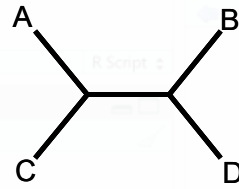
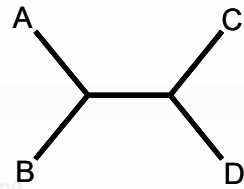
Felsenstein (2004)





# Statistical properties of parsimony

Imagine a binary character on this tree.



There are  $2^4 = 32$  possible site patterns for a binary trait on this tree.

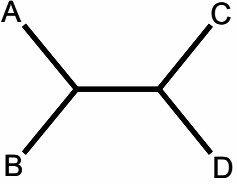
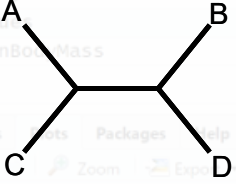
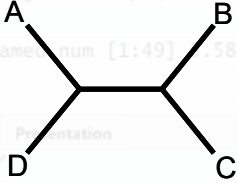
Maximum Parsimony

- So the question is – will picking the tree with the most supporting site patterns (and lowest parsimony score) always yield the correct tree as more data are collected?

Felsenstein (2004)

[illegible]

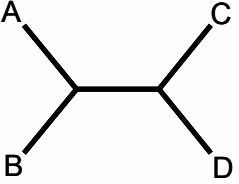
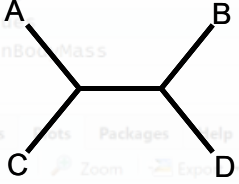
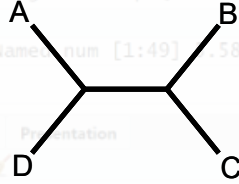
# Statistical properties of parsimony

Pattern ABCD			
0000			
0001			
0010			
0011			
0100			
0101			
0110			
0111			
1000			
1001			
1010			
1011			
1100			
1101			
1110			
1111			

Felsenstein (2004)



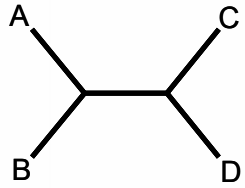
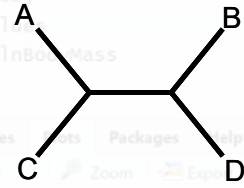
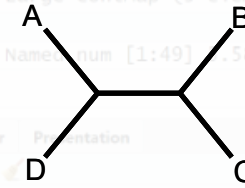
# Statistical properties of parsimony

Pattern ABCD			
0000	0	0	0
0001			
0010			
0011			
0100			
0101			
0110			
0111			
1000			
1001			
1010			
1011			
1100			
1101			
1110			
1111	0	0	0

Felsenstein (2004)

Phylogenetic tree showing relationships between various species, including *O. hemionus*, *O. virginianus*, *R. tarandus*, *A. alces*, *C. dama*, *C. canadensis*, *A. americana*, *A. buselaphus*, *D. lunatus*, *C. taurinus*, *A. melampus*, *H. equinus*, *O. canadensis*, *O. americanus*, *M. kirkii*, *A. cervicapra*, *G. thomsoni*, *G. granti*, *T. oryx*, *B. bison*, *S. caffer*, *G. camelopardalis*, *O. dromedarius*, *L. guanicoe*, *E. burchelli*, *E. caballus*, *E. hemionus*, *D. bicornis*, *C. simum*, *T. bairdi*, *P. leo*, *P. tigris*, *P. pardus*, *A. jubatus*, *C. crocuta*, *H. hyaena*, *V. fulva*, *U. cinereoargenteus*, *C. aureus*, *L. pictus*, *C. latrans*, *C. lupus*, *M. meles*, *M. mephitis*, *P. lotor*, *N. narica*, *U. americanus*, *U. arctos*, and *U. maritimus*.

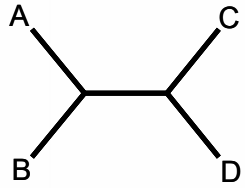
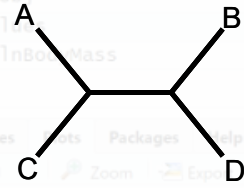
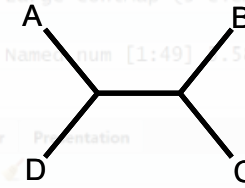
# Statistical properties of parsimony

Pattern ABCD			
0000	0	0	0
0001	1	1	1
0010	1	1	1
0011			
0100	1	1	1
0101			
0110			
0111	1	1	1
1000	1	1	1
1001			
1010			
1011	1	1	1
1100			
1101	1	1	1
1110	1	1	1
1111	0	0	0

Felsenstein (2004)



# Statistical properties of parsimony

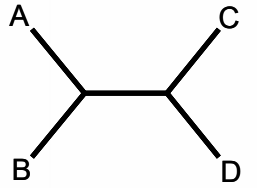
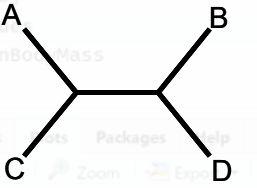
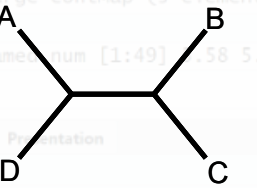
Pattern ABCD			
0000	0	0	0
0001	1	1	1
0010	1	1	1
0011	1	2	2
0100	1	1	1
0101	2	1	2
0110	2	2	1
0111	1	1	1
1000	1	1	1
1001	2	2	1
1010	2	1	2
1011	1	1	1
1100	1	2	2
1101	1	1	1
1110	1	1	1
1111	0	0	0

Felsenstein (2004)





# Statistical properties of parsimony

Pattern ABCD			
0000	0	0	0
0001	1	1	1
0010	1	1	1
0011	1	2	2
0100	1	1	1
0101	2	1	2
0110	2	2	1
0111	1	1	1
1000	1	1	1
1001	2	2	1
1010	2	1	2
1011	1	1	1
1100	1	2	2
1101	1	1	1
1110	1	1	1
1111	0	0	0

xxyy

xyxy

xyyx

xyyx

xyxy

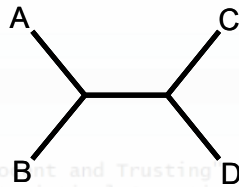
xxyy

Felsenstein (2004)

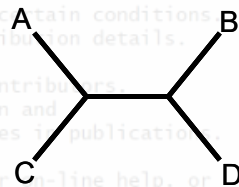


# Statistical properties of parsimony

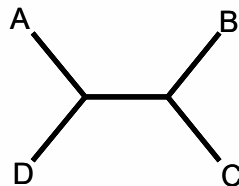
Score from “informative” characters:



$$n_{xxyy} + 2n_{xyxy} + 2n_{xyyx} = \underline{2(n_{xxyy} + n_{xyxy} + n_{xyyx})} - n_{xxyy}$$



$$2n_{xxyy} + n_{xyxy} + 2n_{xyyx} = \underline{2(n_{xxyy} + n_{xyxy} + n_{xyyx})} - n_{xyxy}$$



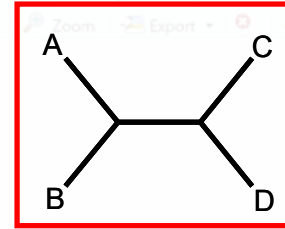
$$2n_{xxyy} + 2n_{xyxy} + n_{xyyx} = \underline{2(n_{xxyy} + n_{xyxy} + n_{xyyx})} - n_{xyyx}$$

Felsenstein (2004)

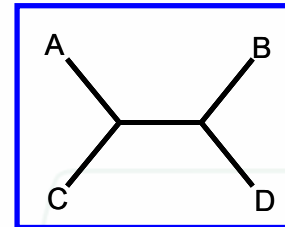
# Statistical properties of parsimony

Observed fractions of site patterns:

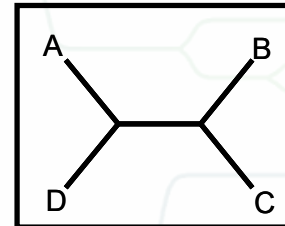
$$f_{xxyy} = n_{xxyy} / N$$



$$f_{xyxy} = n_{xyxy} / N$$



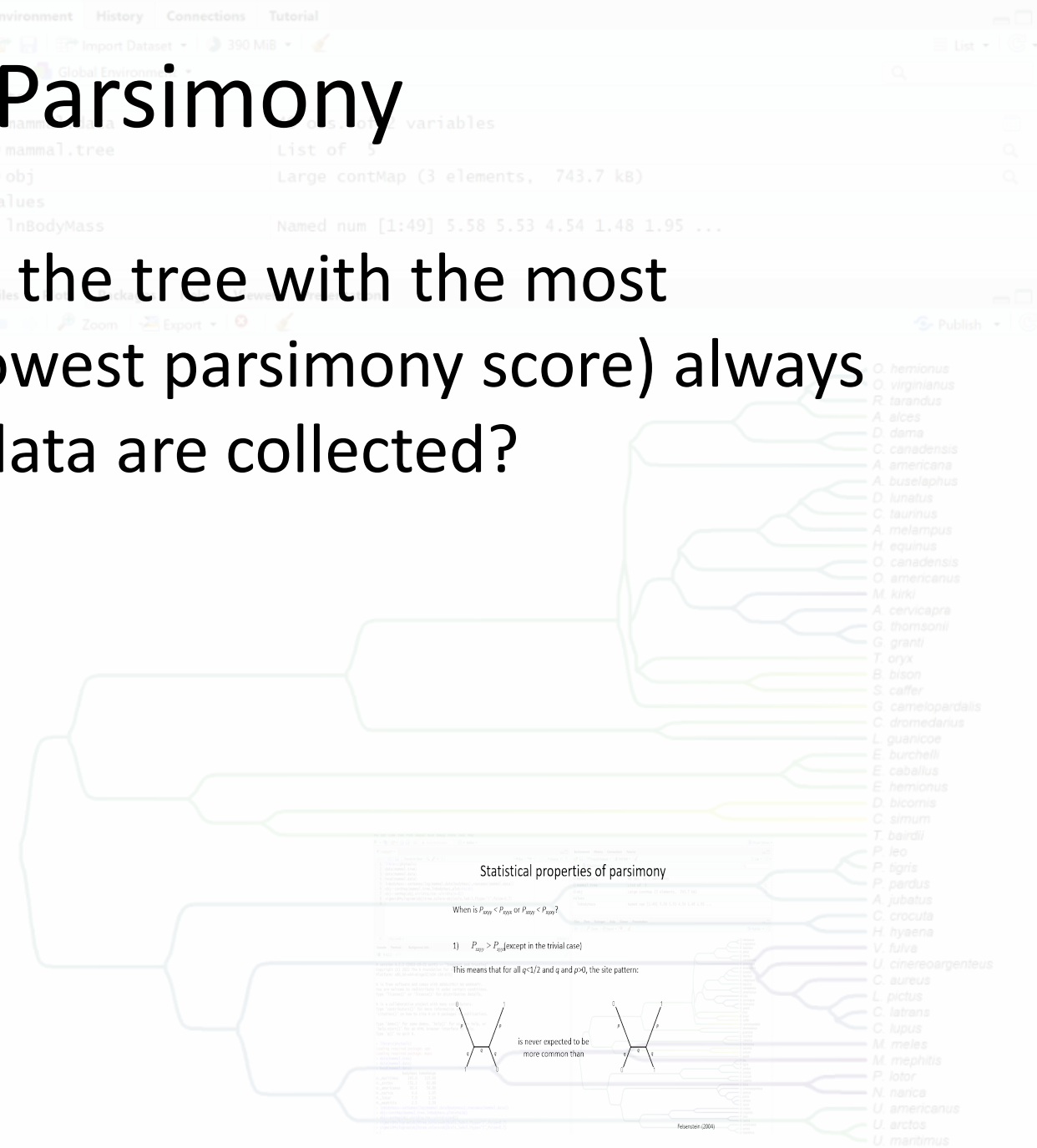
$$f_{xyyx} = n_{xyyx} / N$$



Felsenstein (2004)

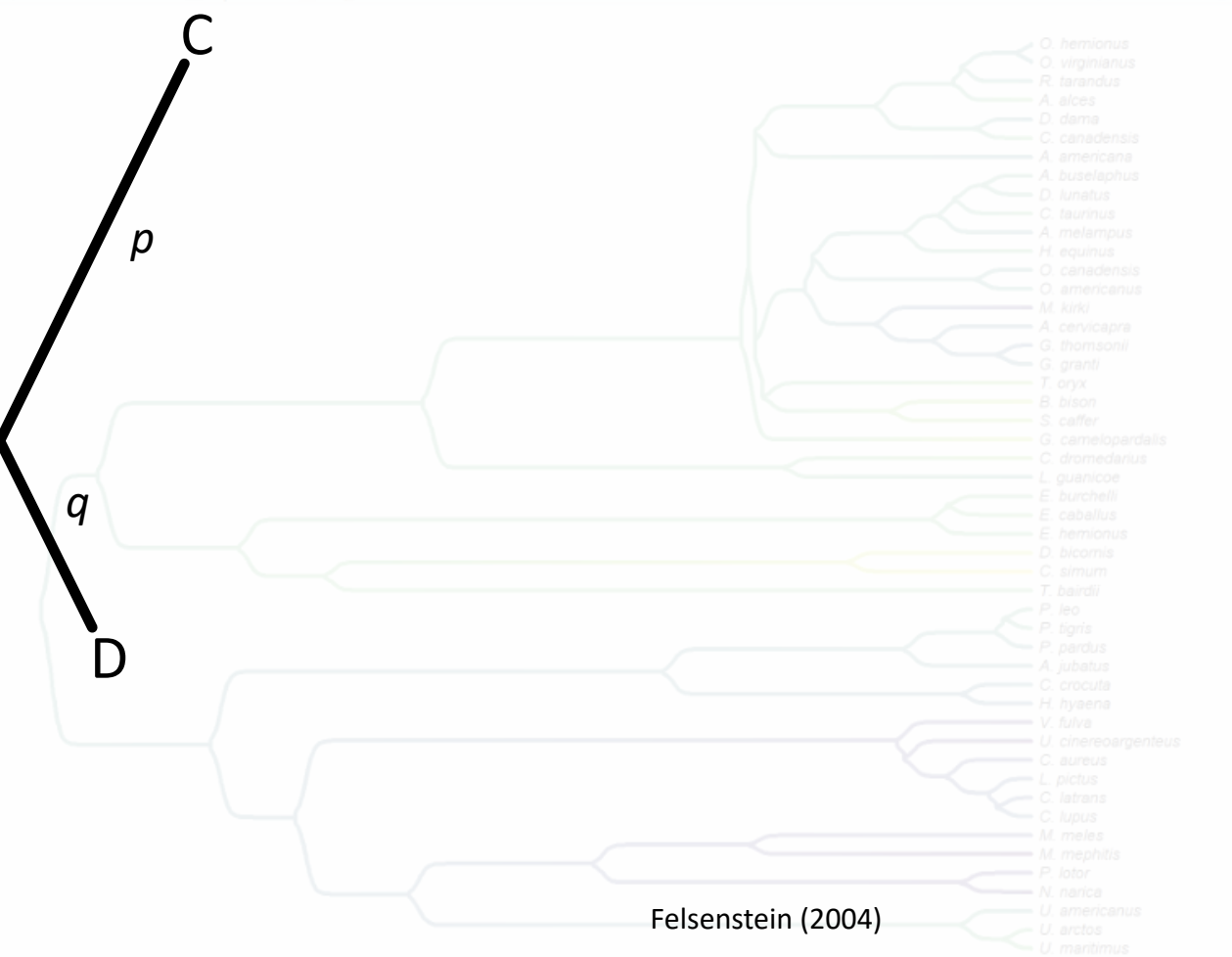
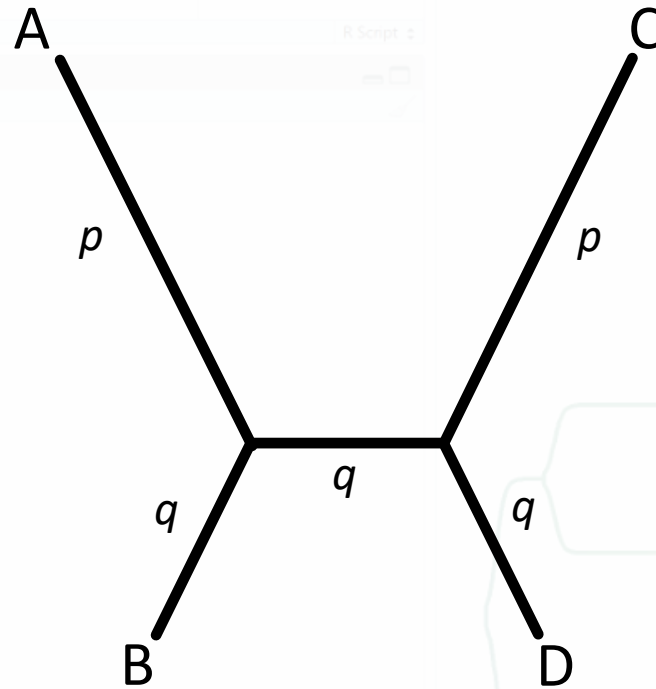
# Maximum Parsimony

- So the question is – will picking the tree with the most supporting site patterns (and lowest parsimony score) always yield the correct tree as more data are collected?



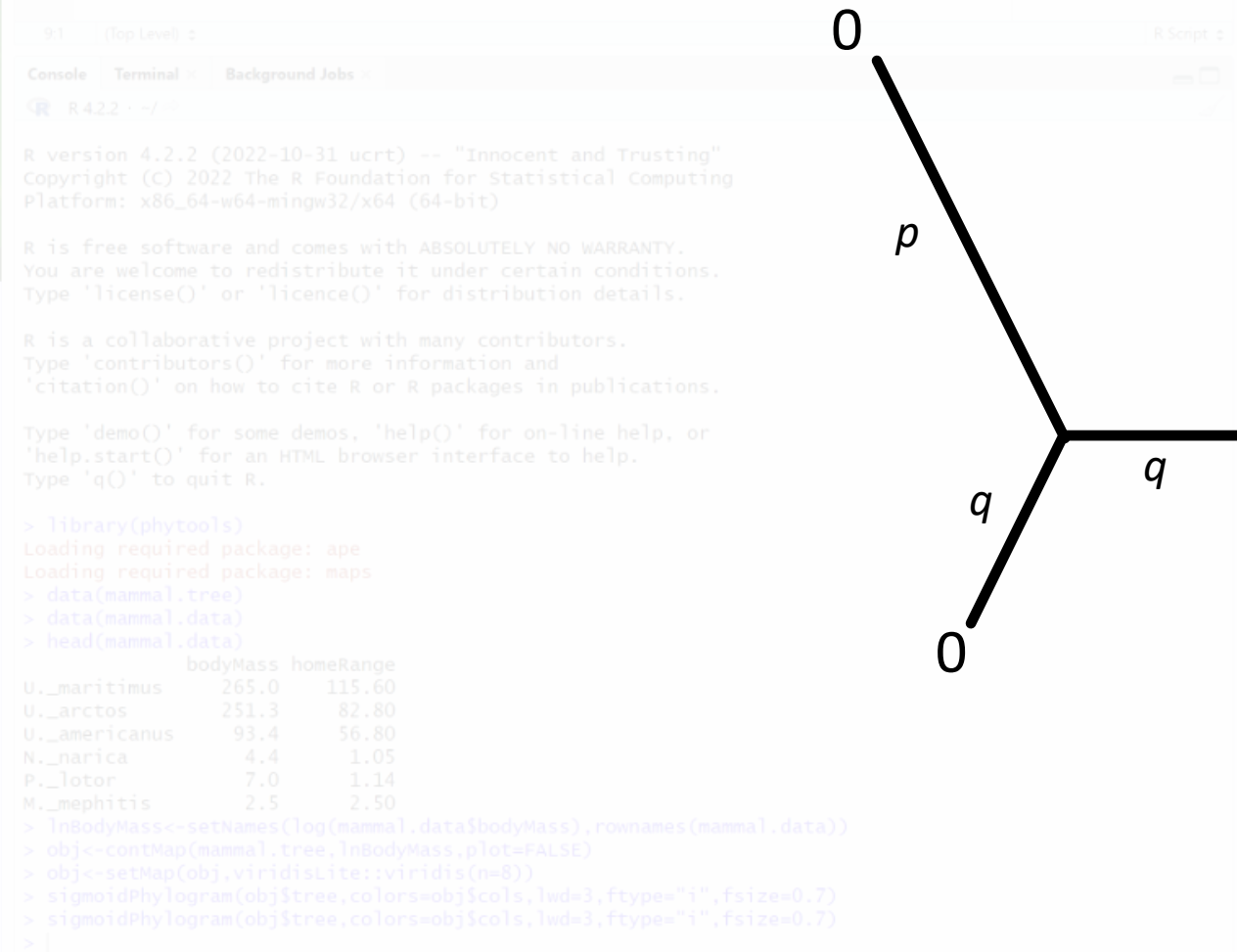
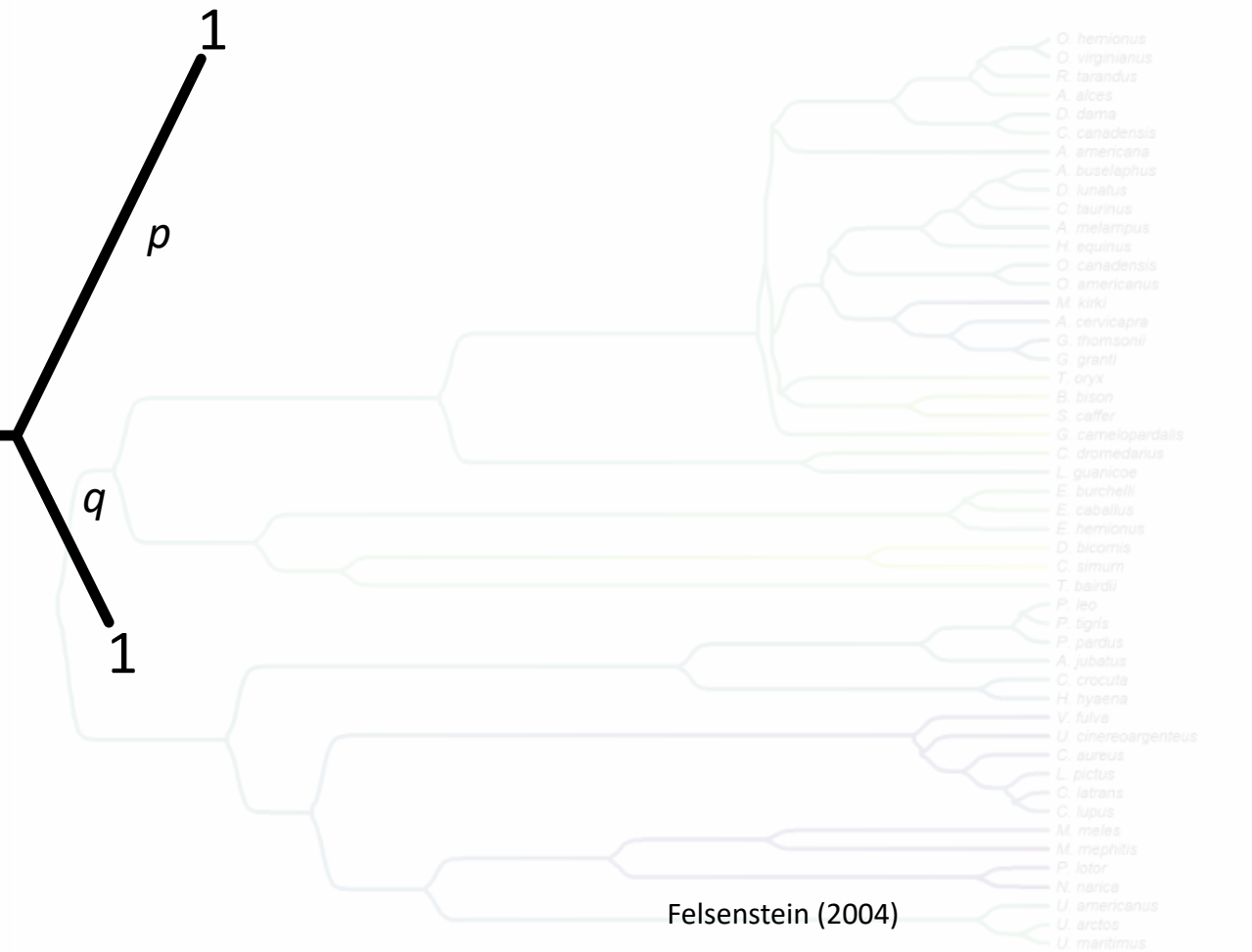
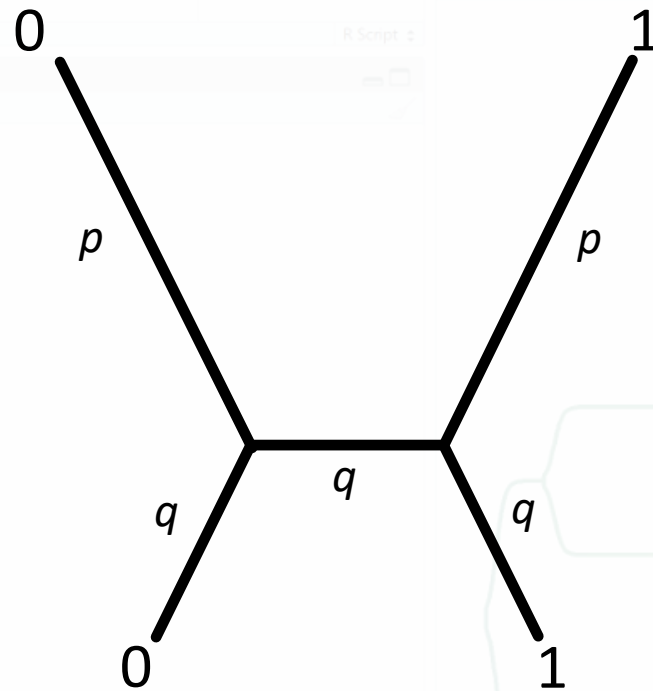
# Statistical properties of parsimony

*Predicted fractions of site patterns:*



# Statistical properties of parsimony

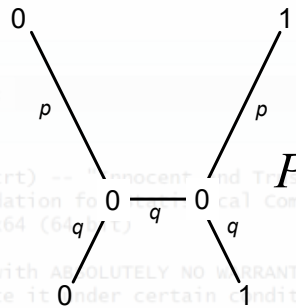
Site pattern 0011





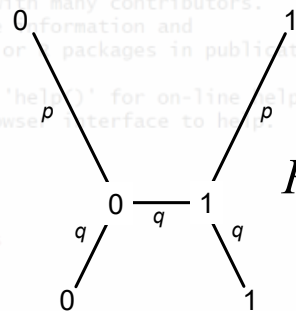
# Statistical properties of parsimony

Site pattern 0011

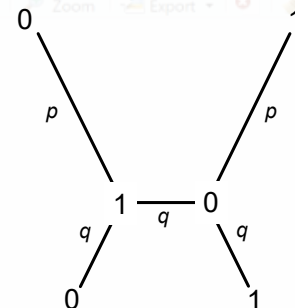


$$P_{0011|00} = \frac{1}{2}(1-p)(1-q)(1-q)pq$$

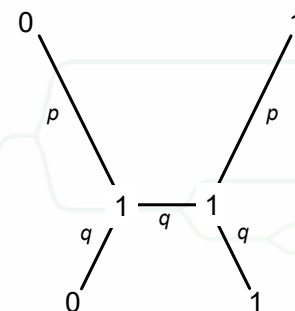
$$= \frac{1}{2}(1-p)(1-q)^2 pq$$



$$P_{0011|01} = \frac{1}{2}(1-p)^2(1-q)^2 q$$



$$P_{0011|00} = \frac{1}{2} p^2 q^3$$



$$P_{0011|00} = \frac{1}{2} pq(1-q)^2(1-p)$$

$$P_{0011} = \frac{1}{2} [(1-p)(1-q)^2 pq + (1-p)^2(1-q)^2 q + p^2 q^3 + pq(1-p)(1-q)^2]$$

$$P_{xxyy} = [(1-p)(1-q)^2 pq + (1-p)^2(1-q)^2 q + p^2 q^3 + pq(1-p)(1-q)^2]$$

Felsenstein (2004)

# Statistical properties of parsimony

Site pattern  $xyxy$ ,  $xyxy$ ,  $xyyx$ :

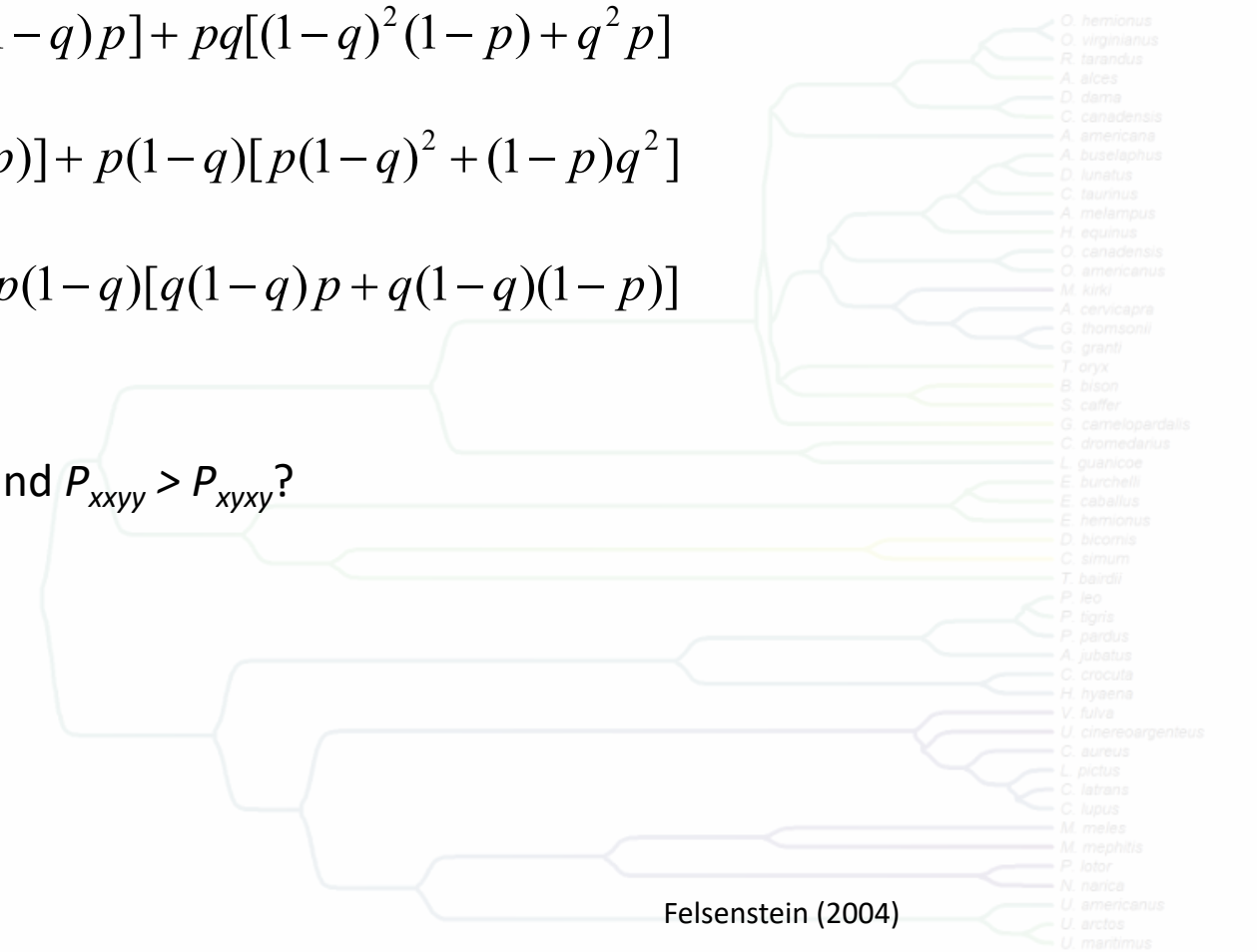
$$P_{xyxy} = (1-p)(1-q)[q(1-q)(1-p) + q(1-q)p] + pq[(1-q)^2(1-p) + q^2p]$$

$$P_{xyxy} = (1-p)q[q(1-q)p + q(1-q)(1-p)] + p(1-q)[p(1-q)^2 + (1-p)q^2]$$

$$P_{xyyx} = (1-p)q[(1-p)q^2 + p(1-q)^2] + p(1-q)[q(1-q)p + q(1-q)(1-p)]$$

So now the question is – is  $P_{xyxy} > P_{xyyx}$  and  $P_{xyxy} > P_{xyxy}$ ?

The answer is **NO!**



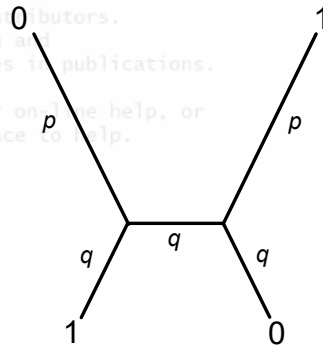
Felsenstein (2004)

# Statistical properties of parsimony

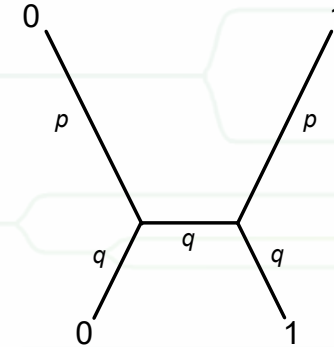
When is  $P_{xxyy} < P_{xyyx}$  or  $P_{xxyy} < P_{xyxy}$ ?

1)  $P_{xxyy} > P_{xyyx}$  (except in the trivial case)

This means that for all  $q < 1/2$  and  $q$  and  $p > 0$ , the site pattern:



is never expected to be more common than



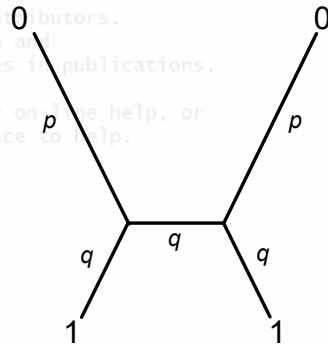
Felsenstein (2004)

# Statistical properties of parsimony

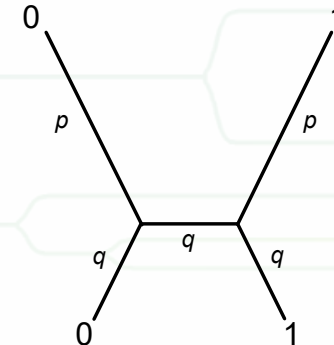
When is  $P_{xxyy} < P_{xyyx}$  or  $P_{xxyy} < P_{xyxy}$ ?

1)  $P_{xxyy} > P_{xyyx}$  (except in the trivial case)

2) However,  $P_{xxyy} > P_{xyxy}$  only for  $q(1-q) > p^2$



will be more common than



whenever  $q(1-q) < p^2$  !!!

Felsenstein (2004)

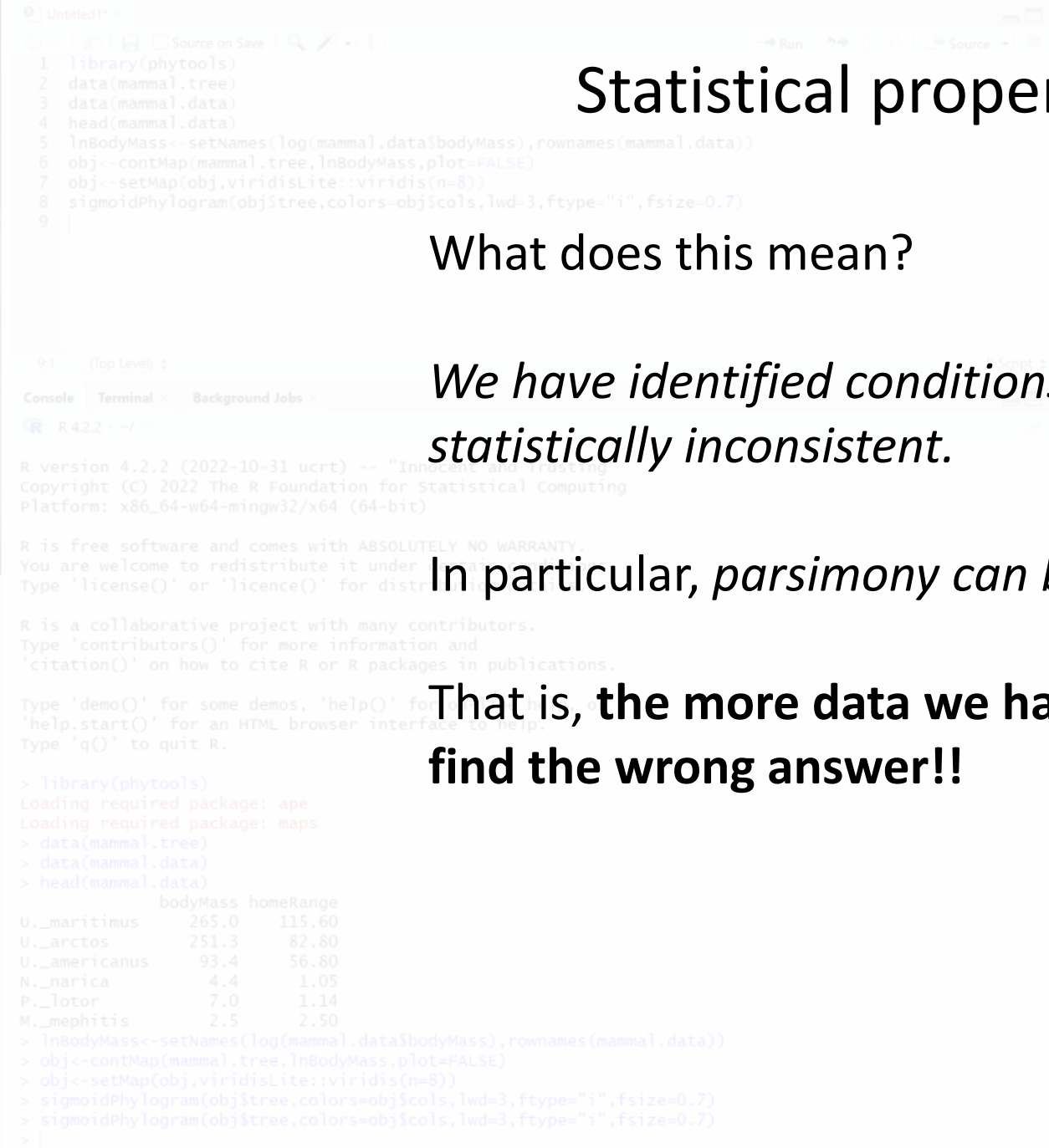
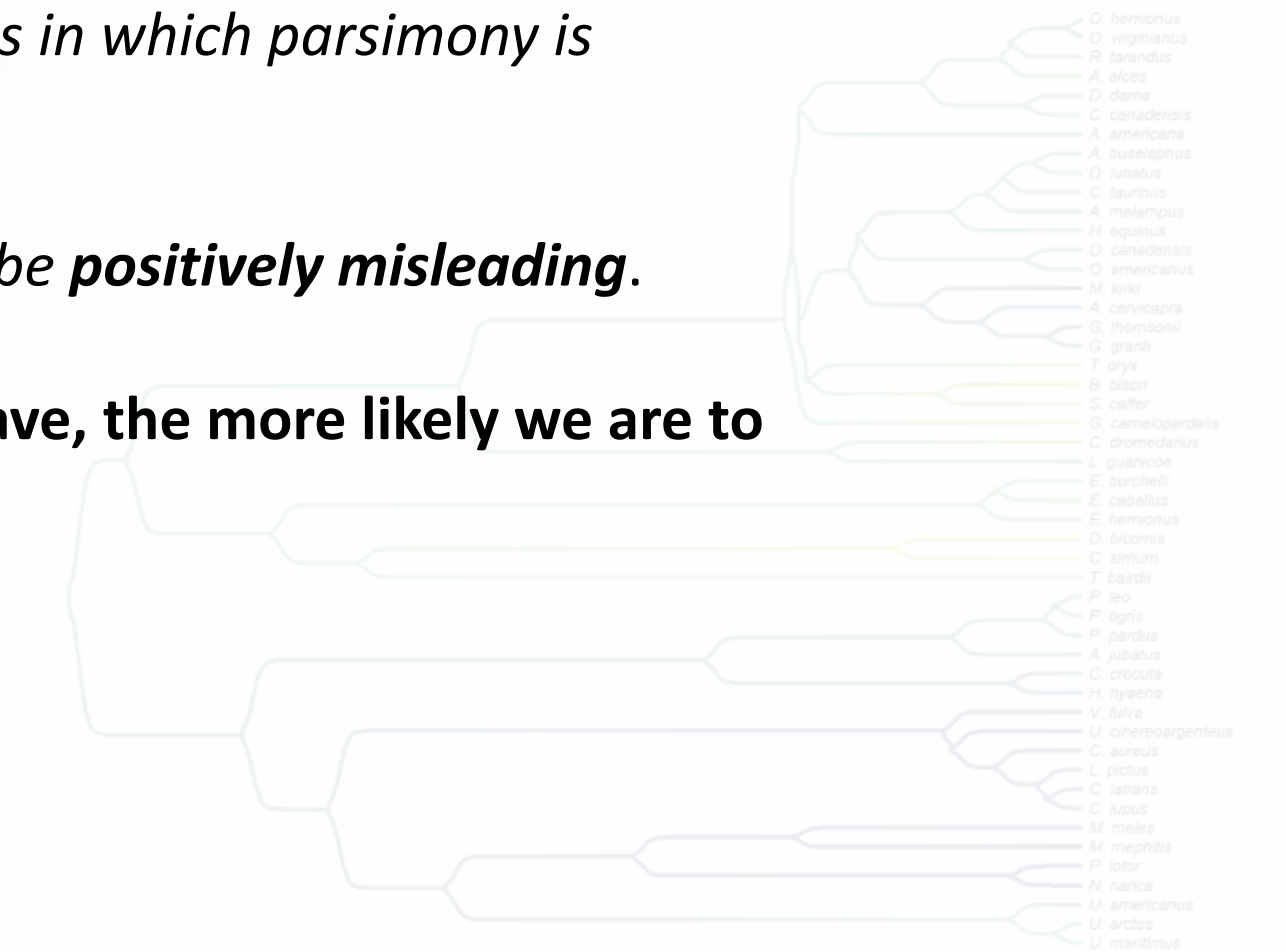
# Statistical properties of parsimony

What does this mean?

*We have identified conditions in which parsimony is statistically inconsistent.*

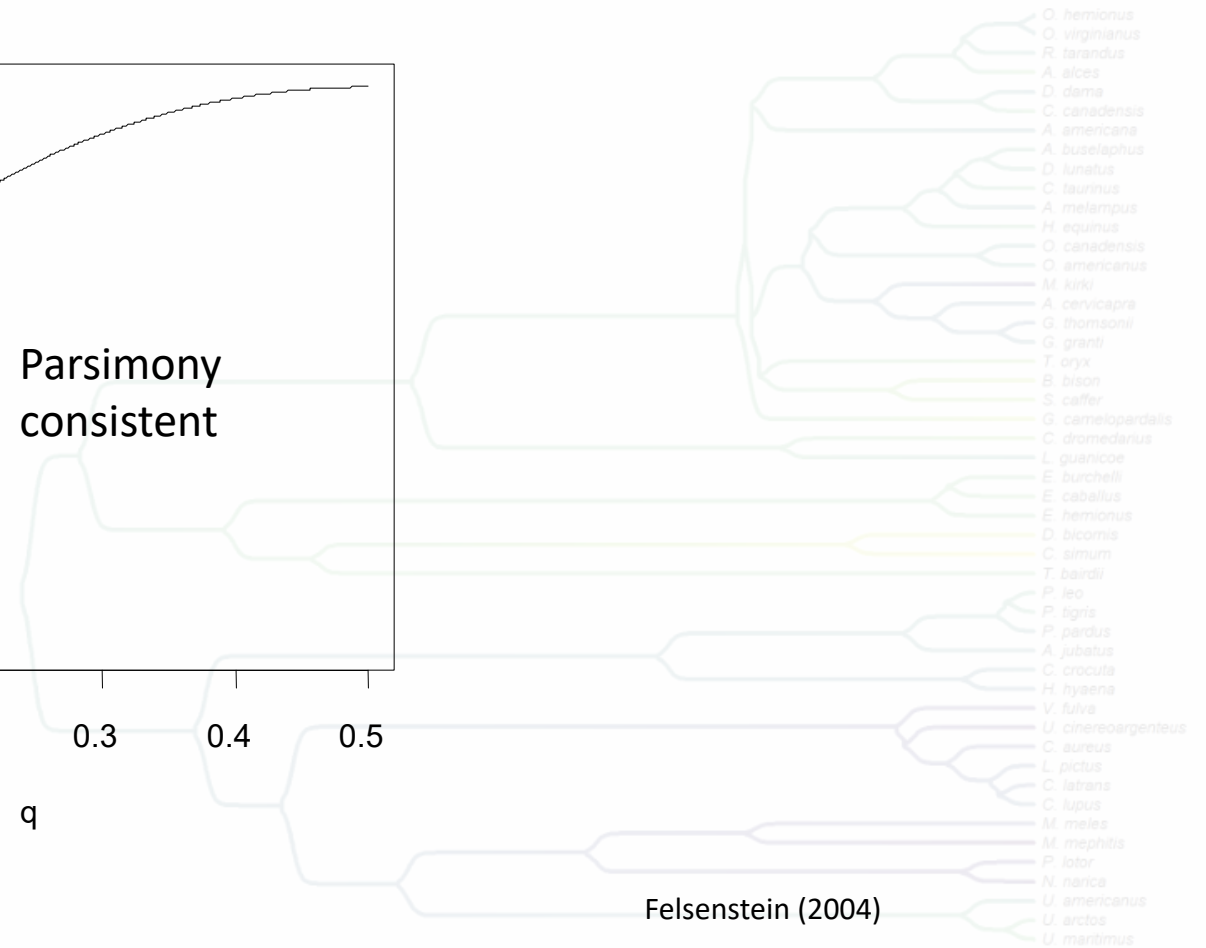
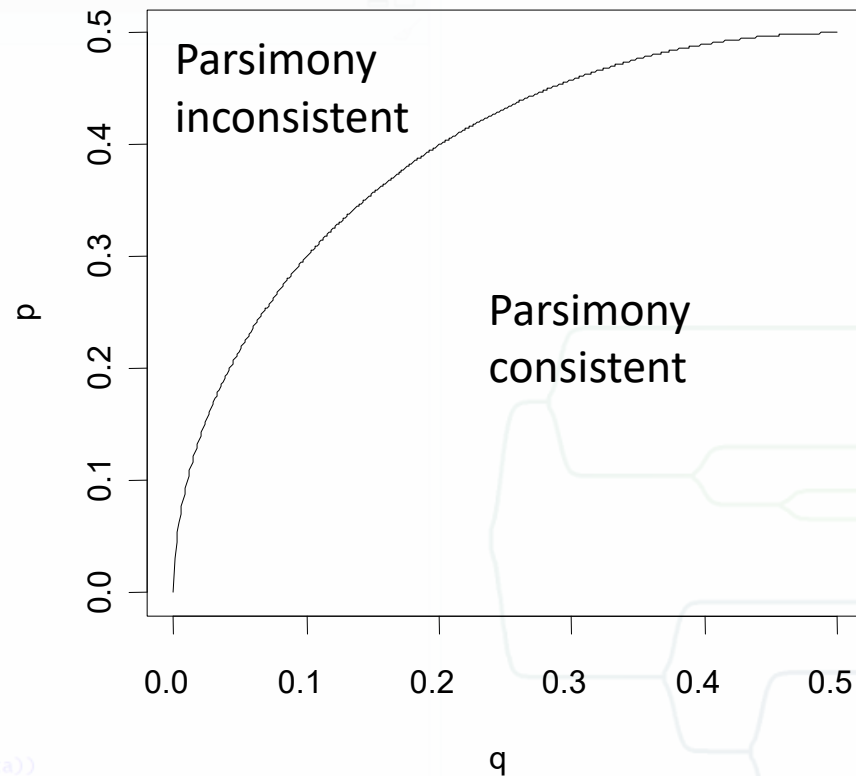
In particular, *parsimony can be **positively misleading**.*

**That is, the more data we have, the more likely we are to find the wrong answer!!**



# Statistical properties of parsimony

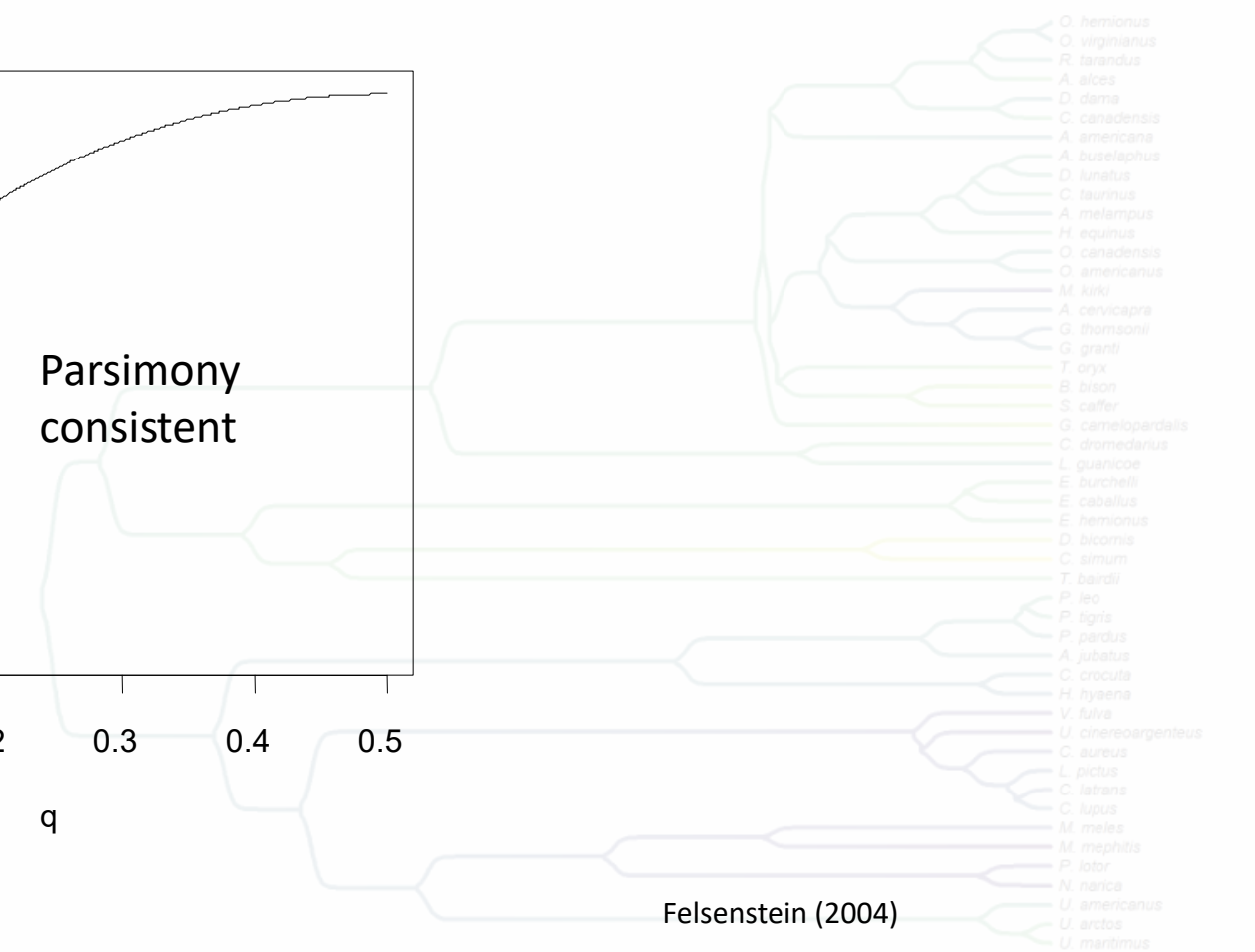
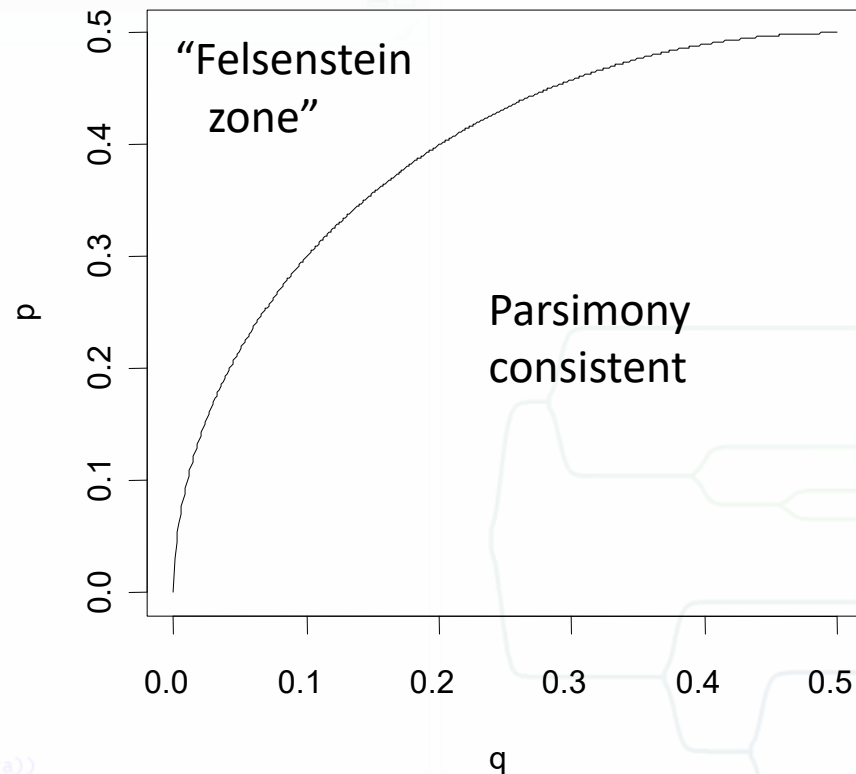
We can plot the regions of consistency and inconsistency.





# Statistical properties of parsimony

We can plot the regions of consistency and inconsistency.



```
library(phytools)
data(mammal.tree)
data(mammal.data)
head(mammal.data)
lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
obj<-setMap(obj,viridisLite::viridis(n=8))
sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftype="i",fsize=0.7)
```

R 4.2.2 (2022-10-31 ucrt) -- "Innocent and Trusting"  
Copyright (C) 2022 The R Foundation for Statistical Computing  
Platform: x86\_64-w64-mingw32/x64 (64-bit)

R is free software and comes with ABSOLUTELY NO WARRANTY.  
You are welcome to redistribute it under certain conditions.  
Type 'license()' or 'licence()' for distribution details.

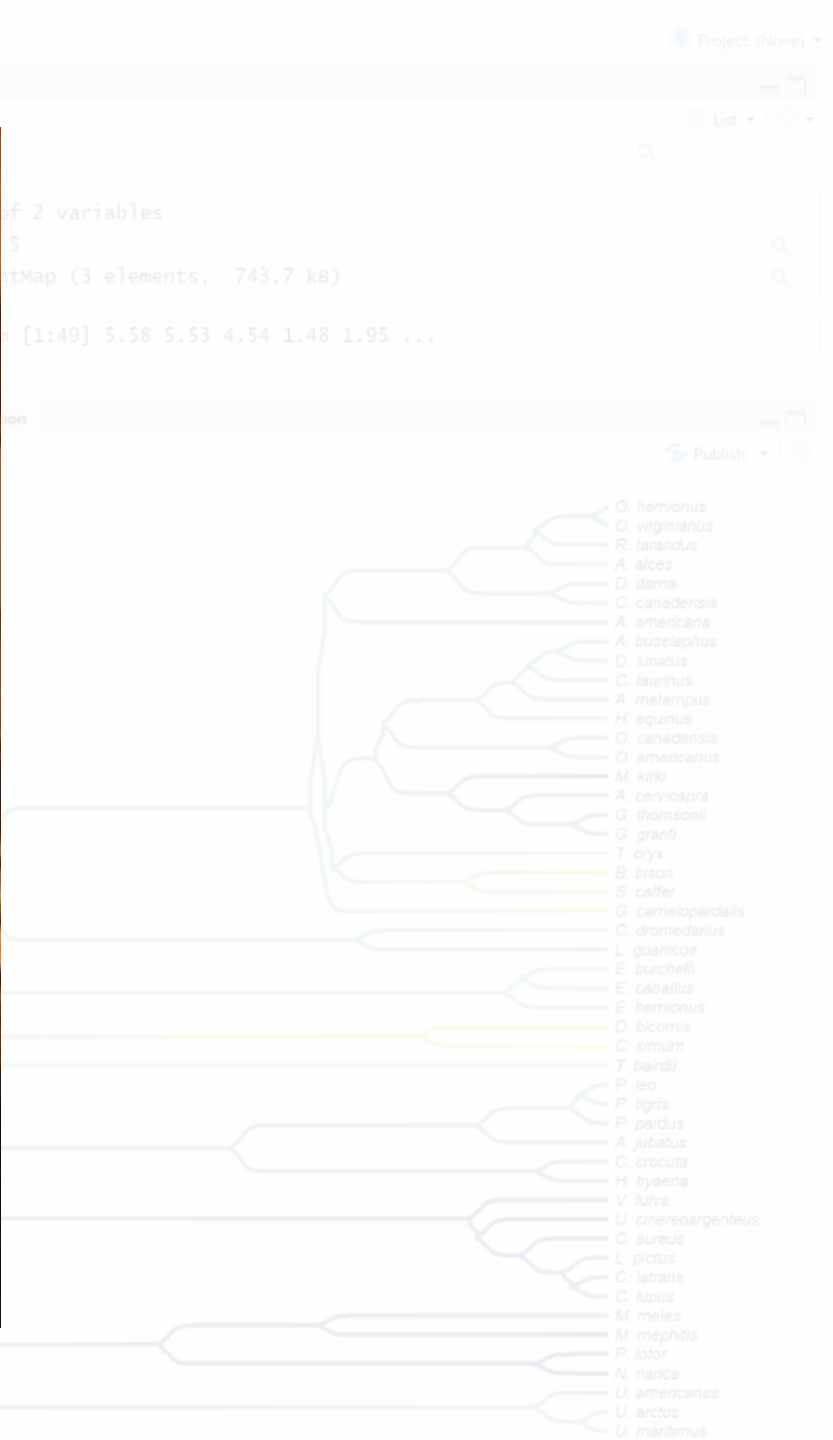
R is a collaborative project with many contributors.  
Type 'contributors()' for more information and  
'citation()' on how to cite R or R packages in publications.

Type 'demo()' for some demos, 'help()' for on-line help, or  
'help.start()' for an HTML browser interface to help.  
Type 'q()' to quit R.

```
> library(phytools)
Loading required package: ape
Loading required package: maps
> data(mammal.tree)
> data(mammal.data)
> head(mammal.data)
  bodyMass homeRange
U._maritimus 265.0    115.60
U._arctos    251.3     82.80
U._americanus 93.4     56.80
N._narica     4.4      1.05
P._lotor      7.0      1.14
M._mephitis   2.5      2.50
> lnBodyMass<-setNames(log(mammal.data$bodyMass),rownames(mammal.data))
> obj<-contMap(mammal.tree,lnBodyMass,plot=FALSE)
> obj<-setMap(obj,viridisLite::viridis(n=8))
> sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftype="i",fsize=0.7)
> sigmoidPhylogram(obj$tree,colors=obj$cols,lwd=3,ftype="i",fsize=0.7)
```

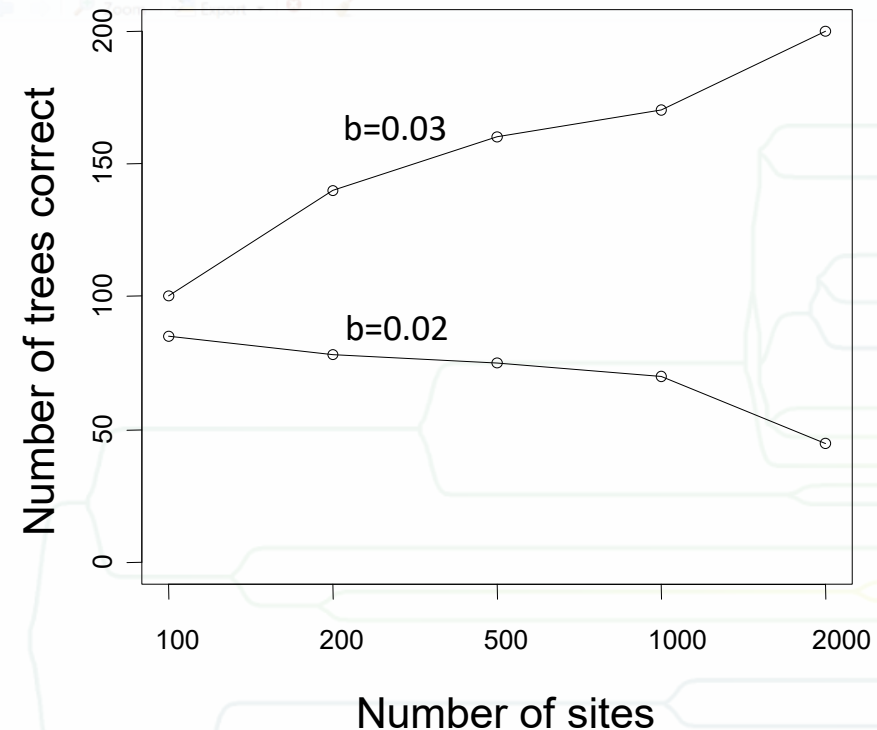
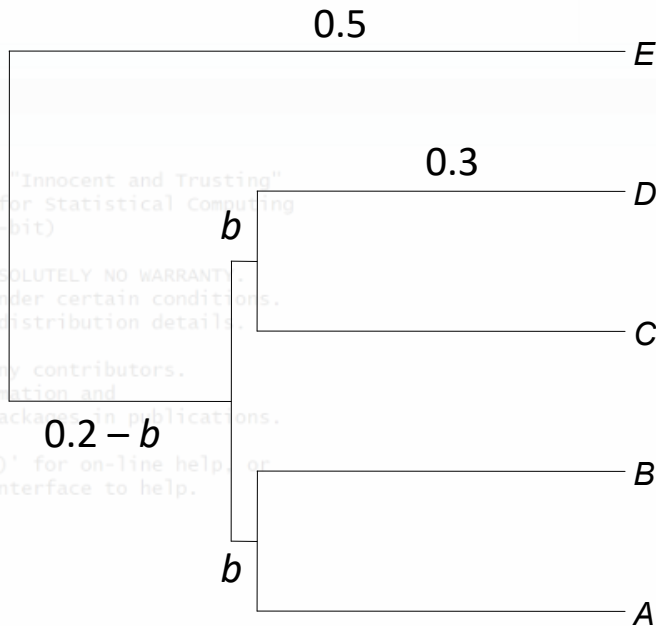


Emily Jane McTavish



# Statistical properties of parsimony

Does a molecular clock guarantee consistency. **NO!!**



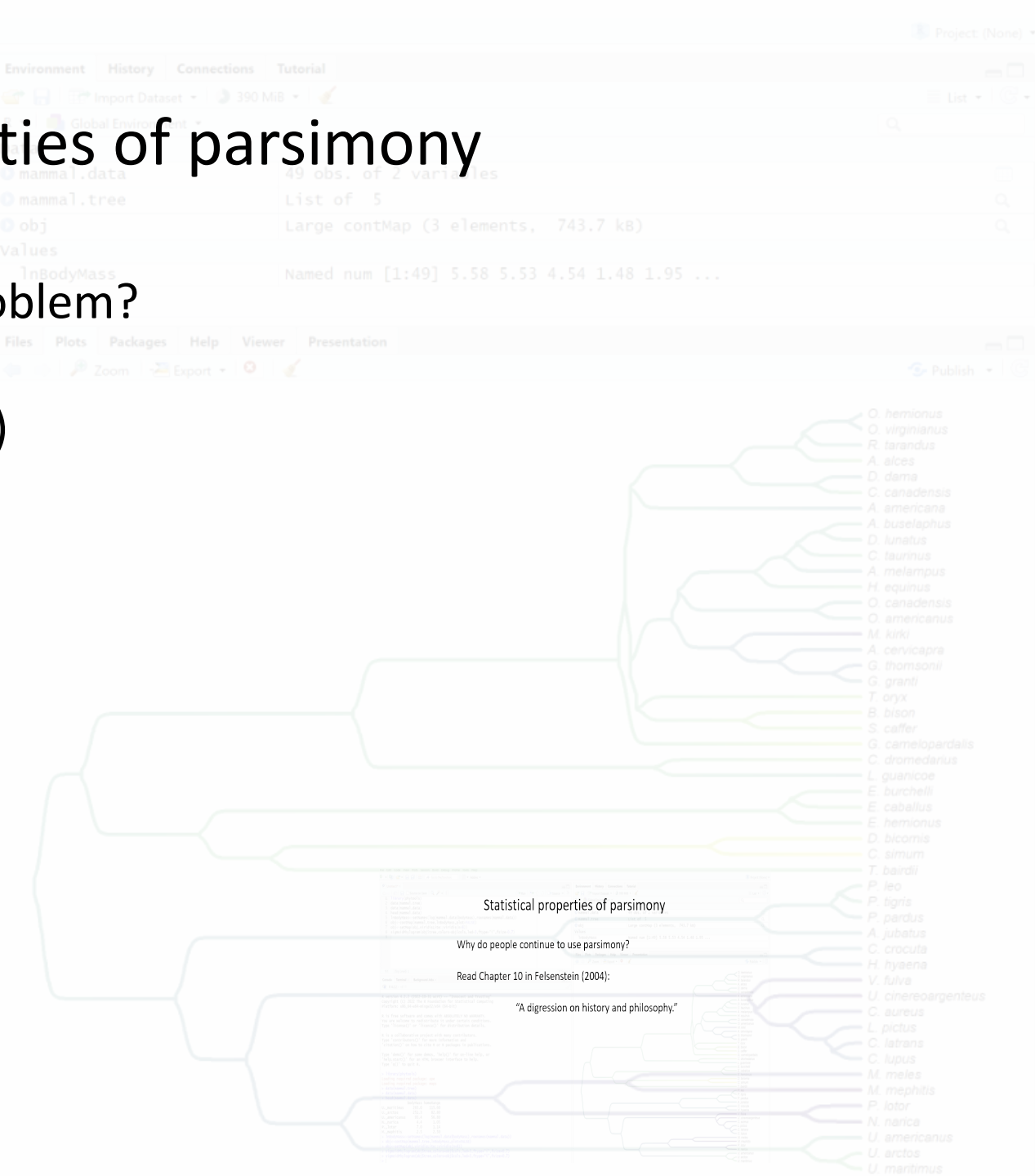
This phenomenon is also called **long branch attraction**.

Felsenstein (2004)

# Statistical properties of parsimony

Is long branch attraction a problem?

Yes!! ([Link to Google scholar.](#))



# Statistical properties of parsimony

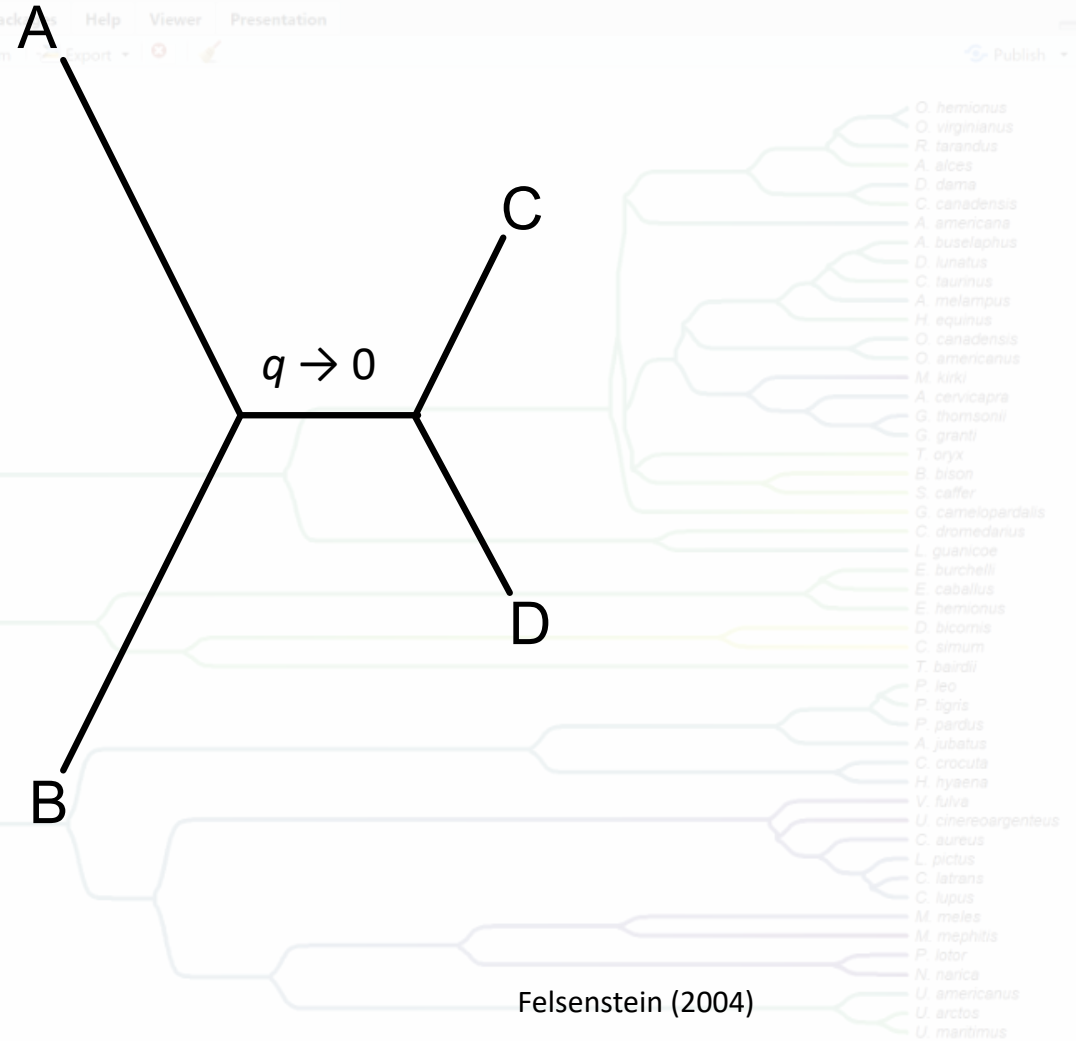
The “Farris zone”

Parsimony more efficient at resolving correct topology in this case.

Why?

Parsimony will group A & B because of chance similarity acquired along their long branches, rather than due to true homology.

Proof: as  $q \rightarrow 0$ , parsimony will continue to find the “correct” topology.





# Statistical properties of parsimony

Why do people continue to use parsimony?

Read Chapter 10 in Felsenstein (2004):

“A digression on history and philosophy.”

