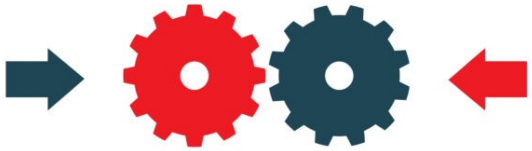
BIOINFORMATICS

BIOLOGY

We will draw from our understanding of DNA, RNA, and proteins and how they contribute to cellular processes

COMPUTER SCIENCE

We will learn to using coding and command line programs to analyze biological data



INFORMATION ENGINEERING

We will learn how to store, process, and share large datasets

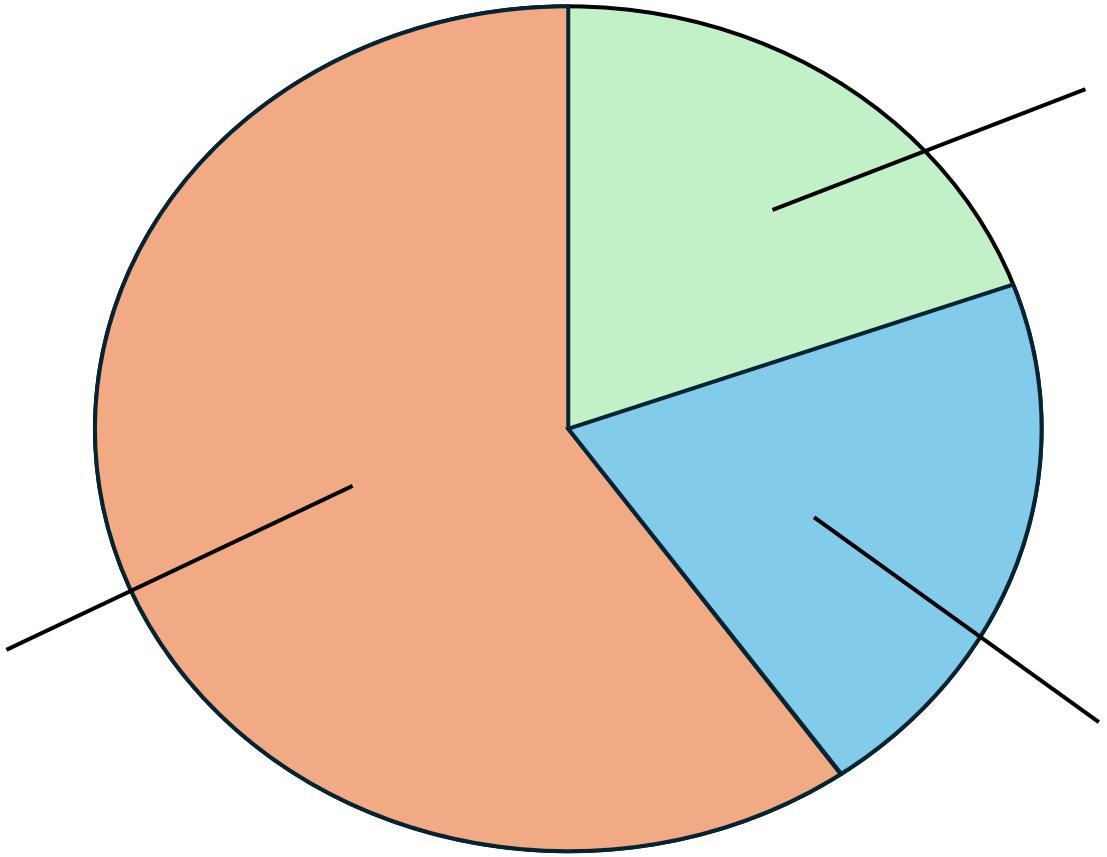
MATHEMATICS

We will employ algorithms that rely heavily on mathematics

STATISTICS

We will learn to use statistics to help us understand the patterns and trends in our analyses

1000 total points



Tutorial assignments:

- 1 per week
- 20 points each = 200 total points
- Will accompany in-class tutorials/exercises
- Assigned Tuesday of each week

Weekly project write-ups:

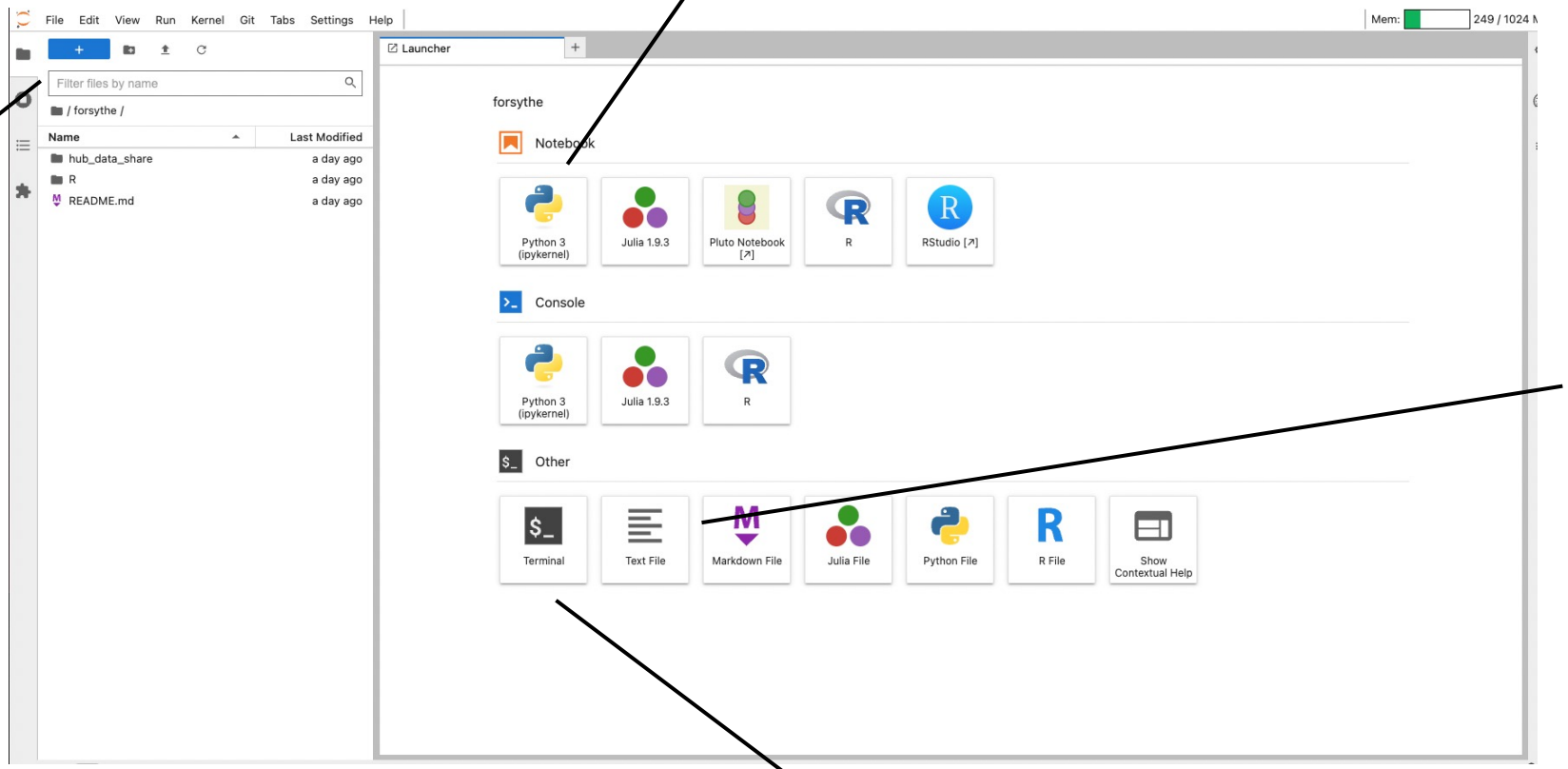
- 1 per week for 6 weeks (week 3-8)
- 100 points each = 600 total points
- A larger applied analysis of a real biological dataset
- Assigned Thursday of every week

Final project:

- Choose from projects covered in week 9
- 200 total points
- Will yield a publication-quality analysis and figure(s)

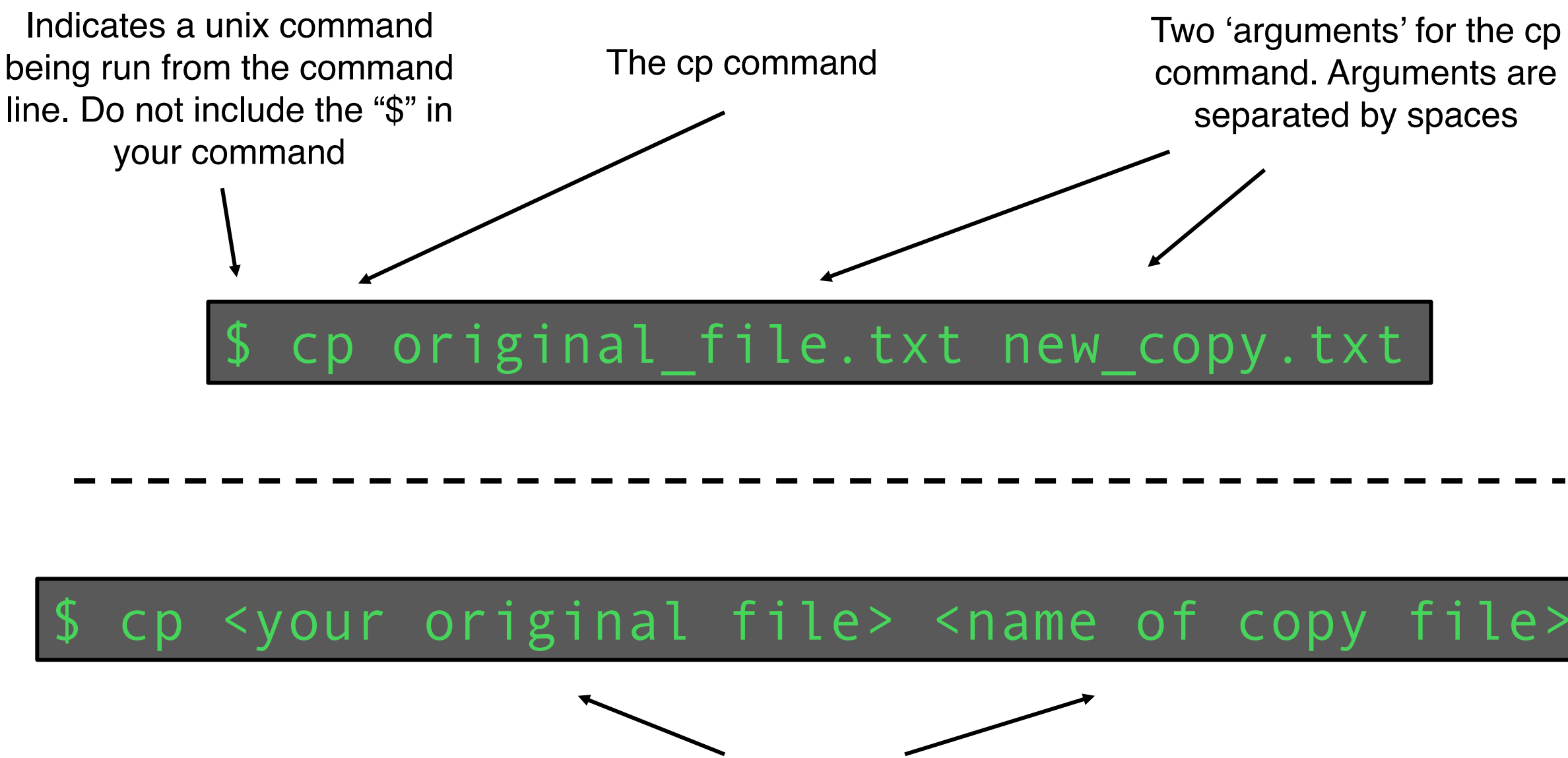
Opens a python notebook for testing python code

Opens a new ‘launcher’ window, which can be used to open different types of tools

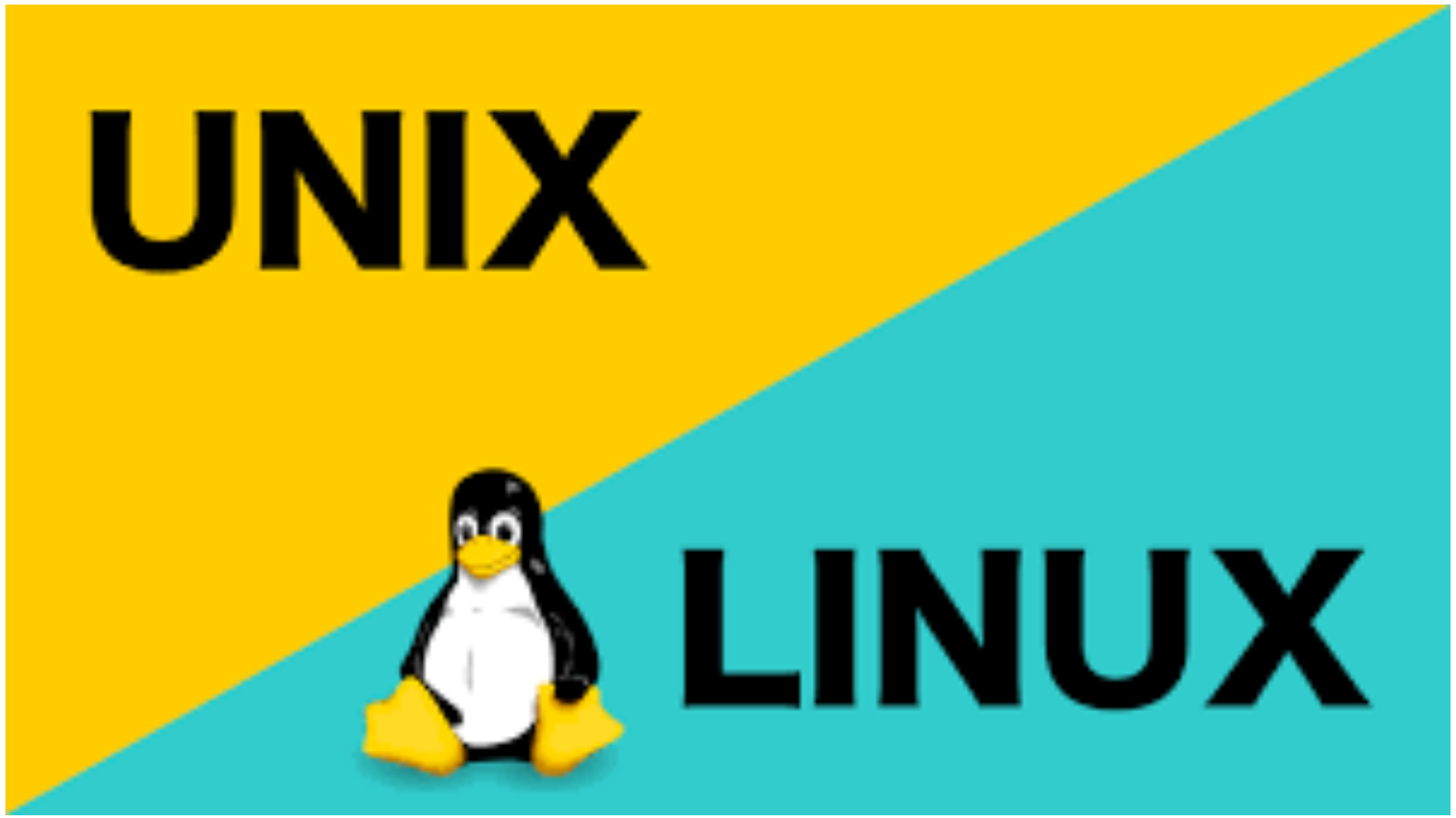


Opens a text file for creating scripts (e.g. python scripts)

Opens a command line terminal for navigating the file system or running linux/unix commands


Indicates a unix command being run from the command line. Do not include the “$” in your command
The cp command
Two ‘arguments’ for the cp command. Arguments are separated by spaces
$ cp original_file.txt new_copy.txt
$ cp <your original file> <name of copy file>
Text surround by < > means this is info you need to fill in yourself. Do not include the “<” and “>” in your command

BioInformatics with python™


UNIX
LINUX

How do computers work?

Computer: an electronic device for storing and processing **data**

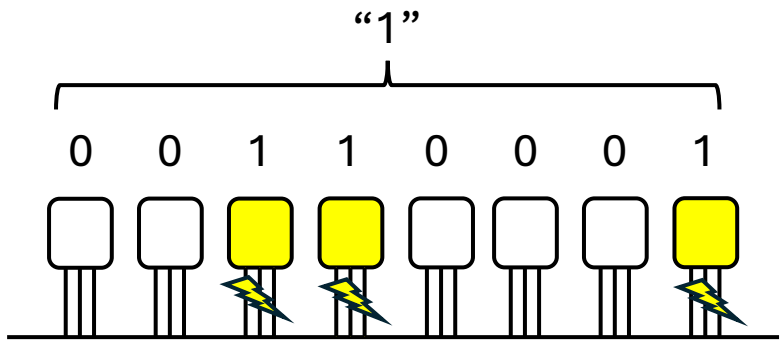
BINARY CODE CONVERTER			
A	01000001	S	01010011
B	01000010	T	01010100
C	01000011	U	01010101
D	01000100	V	01010110
E	01000101	W	01010111
F	01000110	X	01011000
G	01000111	Y	01011001
H	01001000	Z	01011010
I	01001001	0	00110000
J	01001010	1	00110001
K	01001011	2	00110010
L	01001100	3	00110011
M	01001101	4	00110100
N	01001110	5	00110101
O	01001111	6	00110110
P	01010000	7	00110111
Q	01010001	8	00111000
R	01010010	9	00111001

1 bit

8 bits = 1 byte

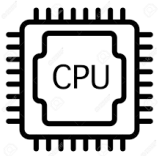
Why is computer data stored in binary?

- Data are stored by electrically manipulating physical components called **transistors**



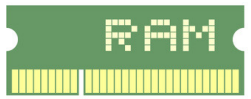
https://onlinetexttools.com/convert-text-to-binary

Hardware components of a simplified computer



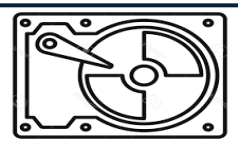
Central processing unit

- Contains 1 or many 'cores'
- Executes processes (*e.g. calculate 1+2*)



RAM / 'Memory'

- Temporary storage during processing
- *(e.g. 00110001 + 00110010 = 00110011)*



Harddrive/'Disk'/'Storage'

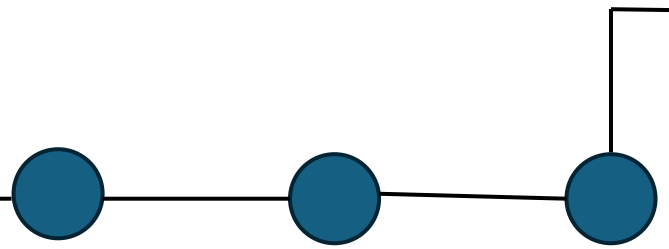
- Data is saved/'written' to disk for long-term storage
- e.g. *00110011 (i.e. "3")*

Remote servers and ‘the cloud’

All data need to be physically stored somewhere:

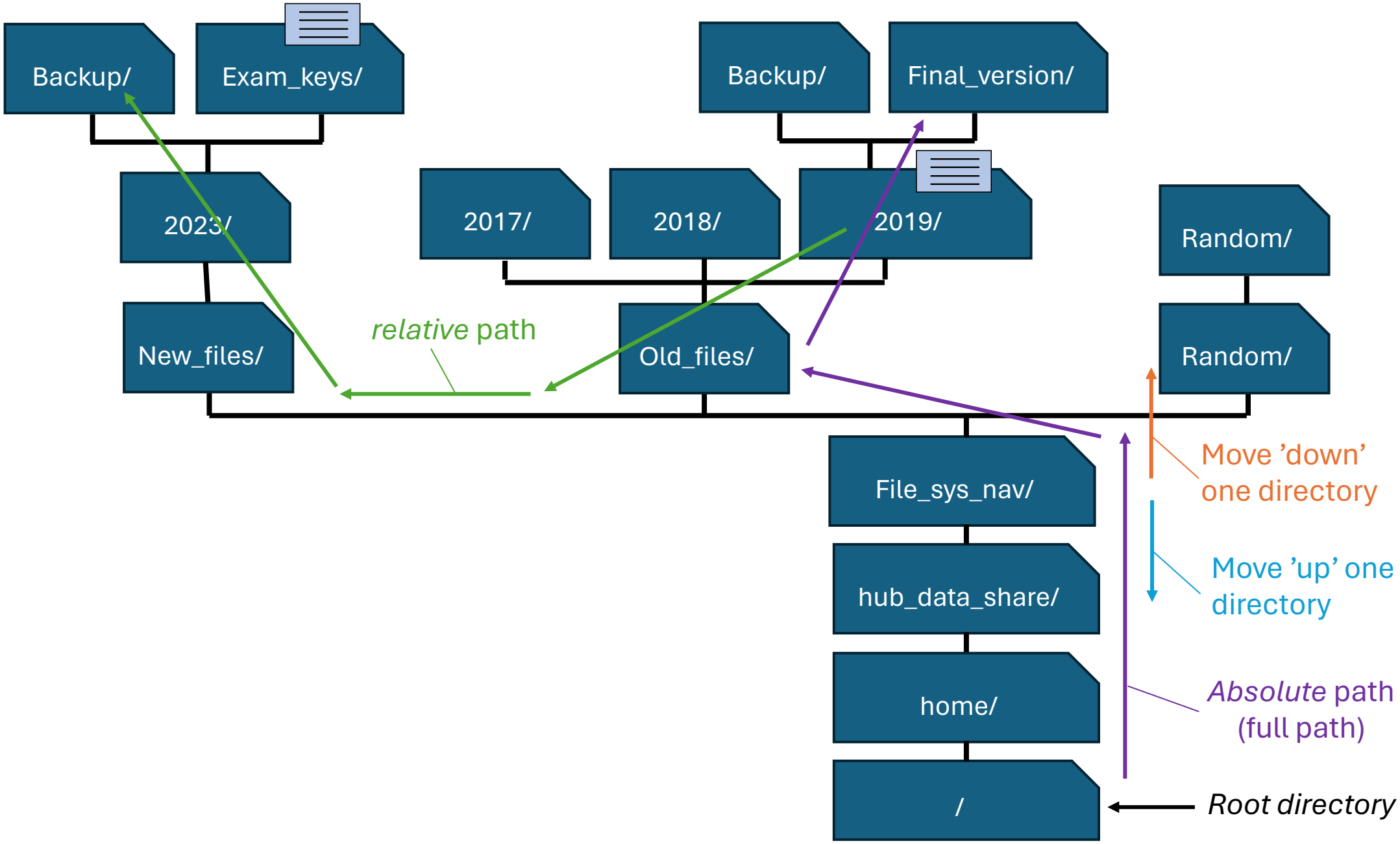
- The data on the internet are physically stored a series of interconnected servers
- When you visit a website, your devices pings a message to a server somewhere in the world and requests access to the data stored there

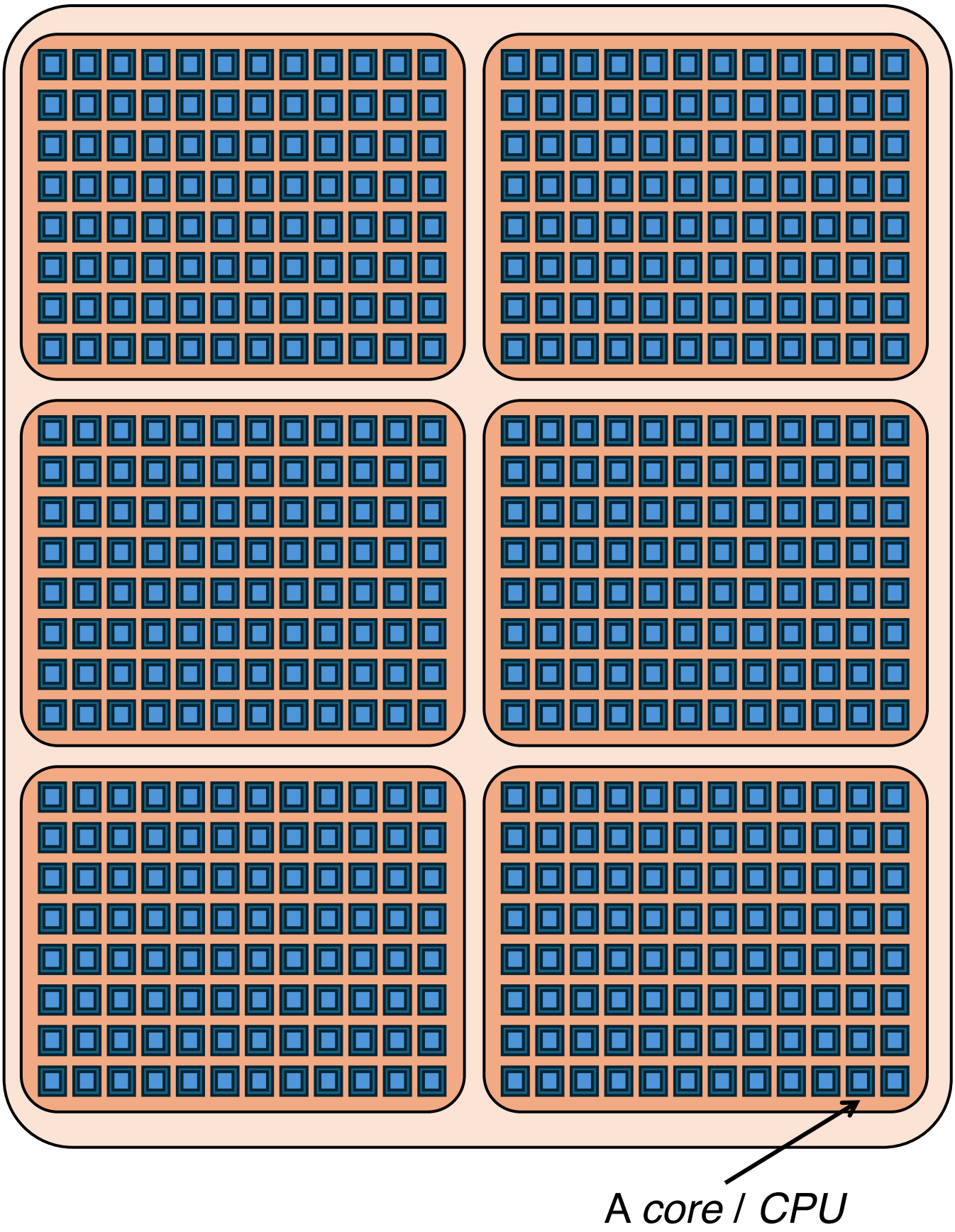
Personal computer: your ‘local machine



Server (‘remote machine’): big computer with hundreds of CPUs

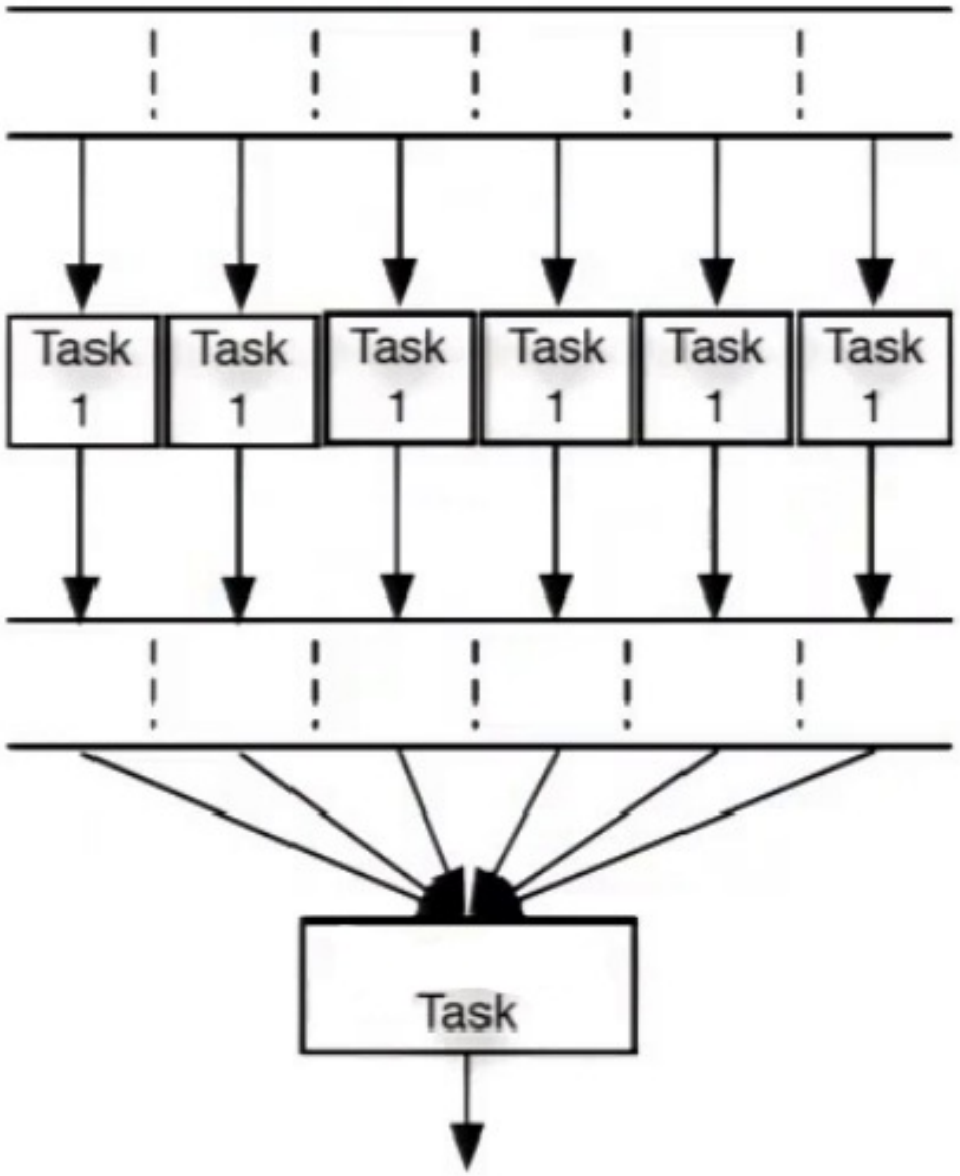
An example file system



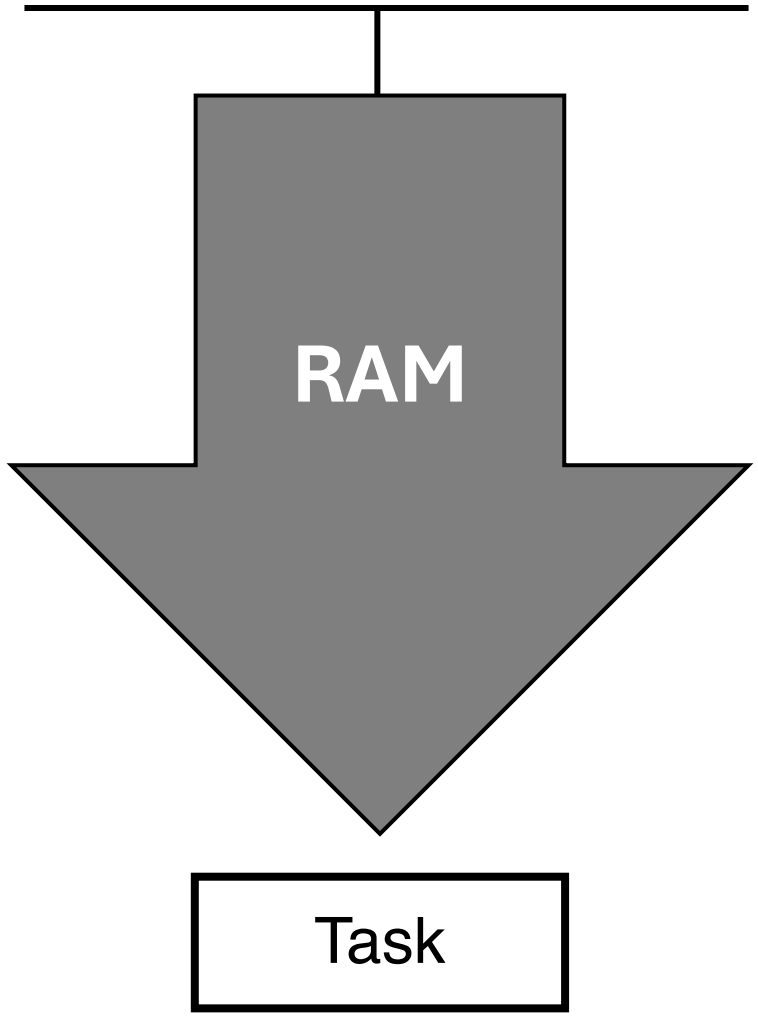

A core / CPU
A node
A server / HPC / cluster


A node

Parallel computing



Memory-intensive computing



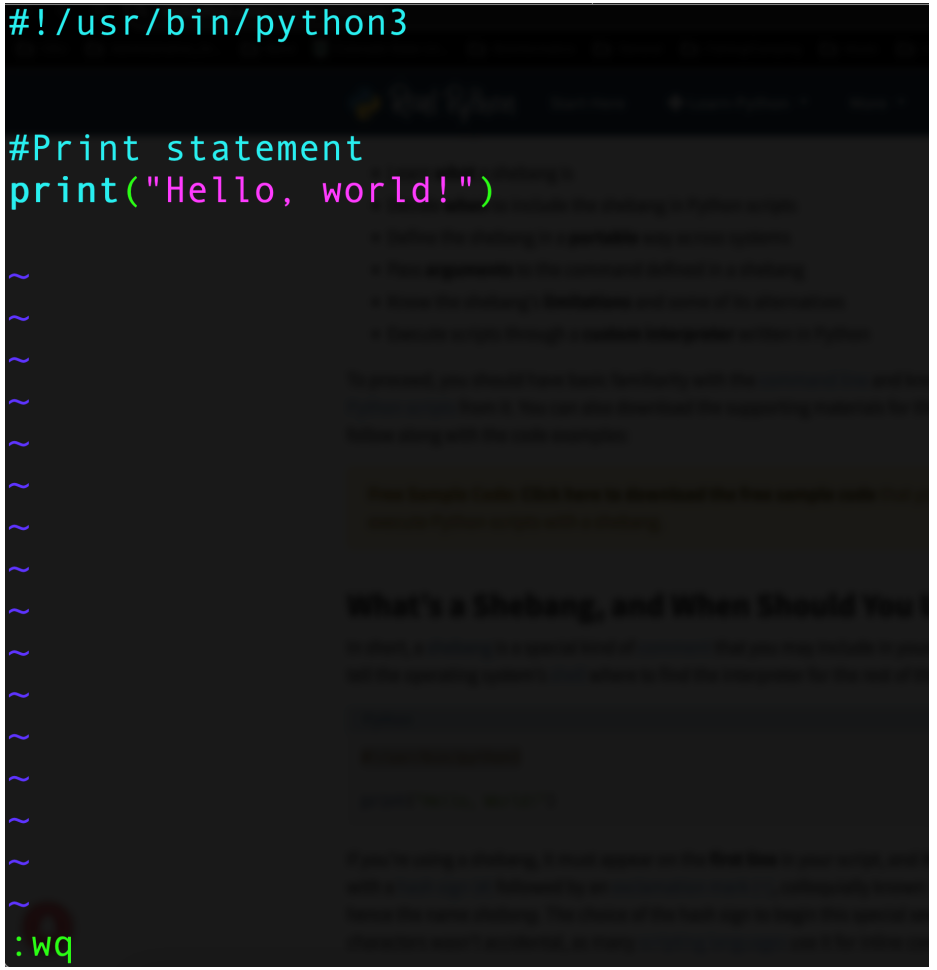
Text editing with *vim*

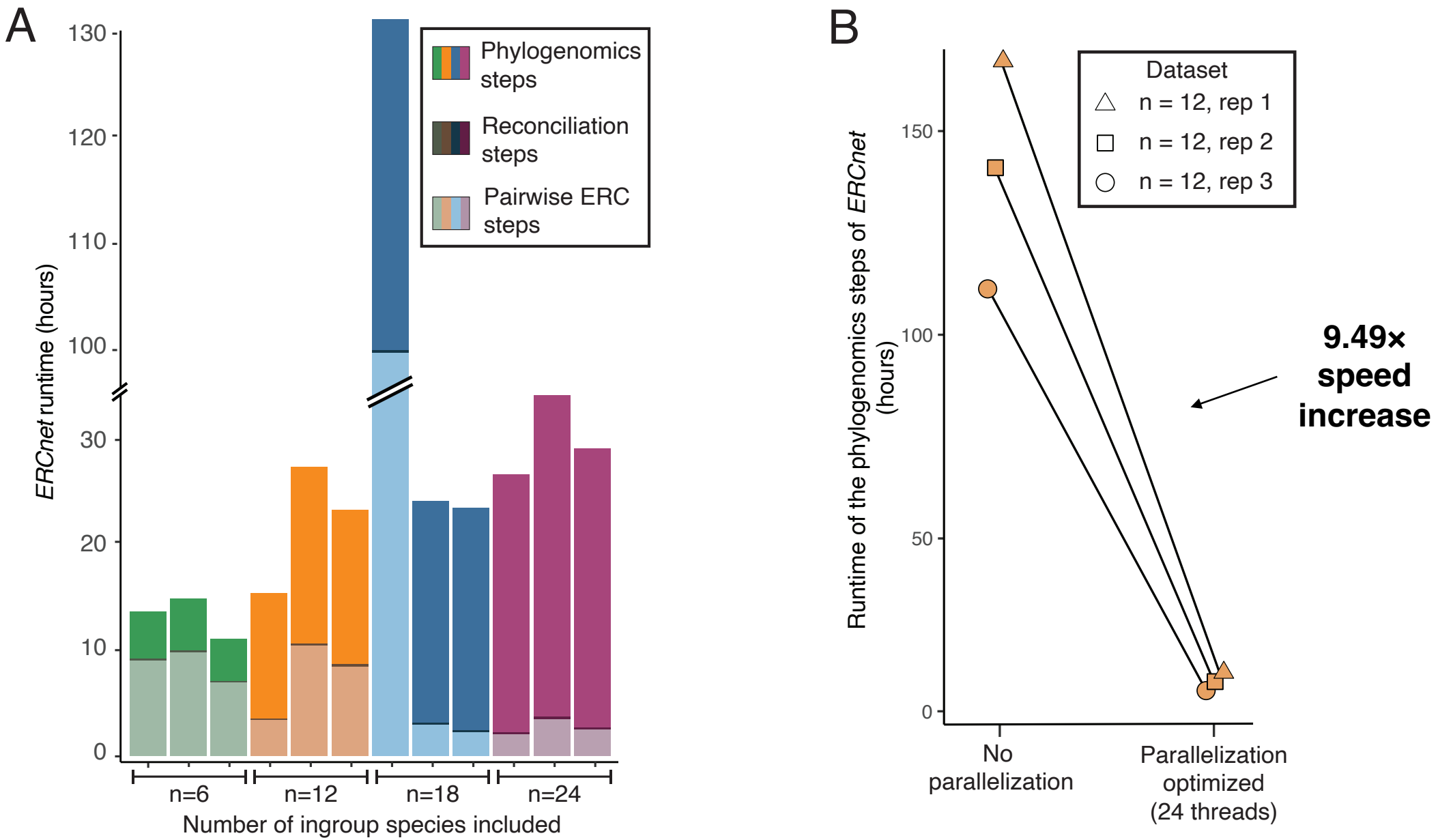
Insert mode: type text into your document

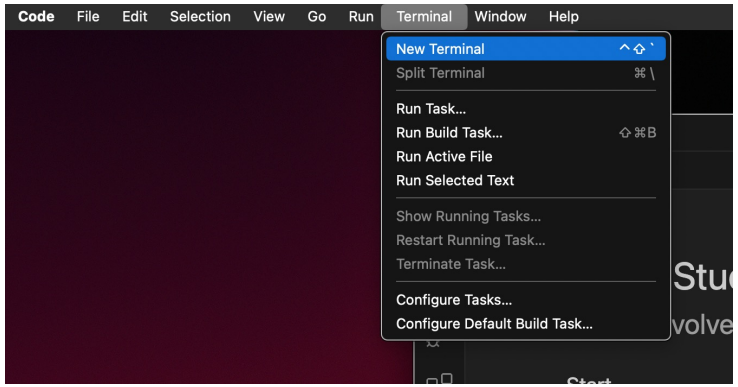
Command mode: enter ":wq" command to write and save

Press "i"

Press "Esc"

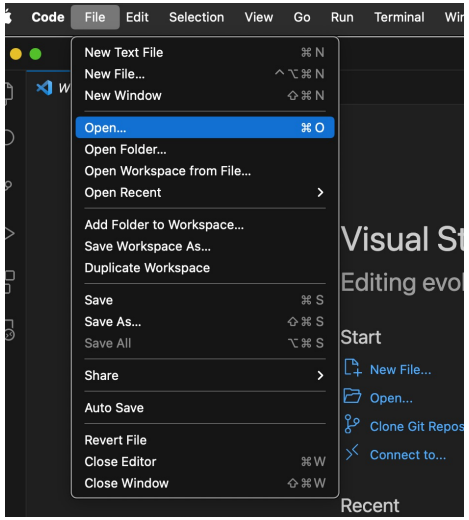


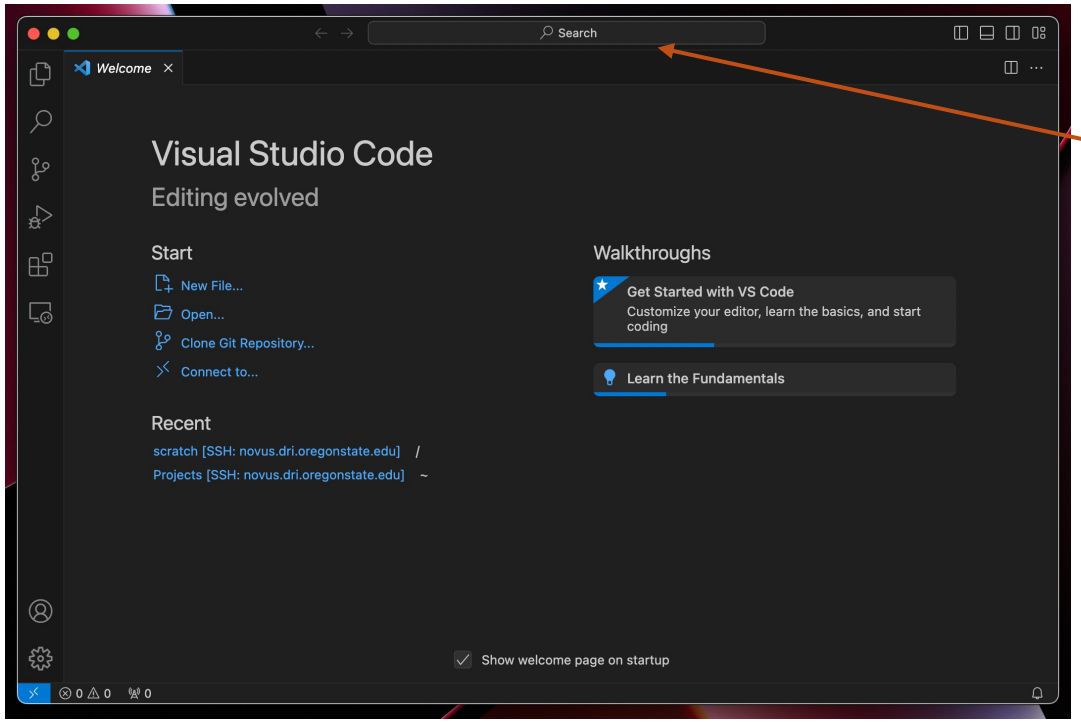

A
130
120
110
100
30
20
10
0
ERCnet runtime (hours)
Phylogenomics steps
Reconciliation steps
Pairwise ERC steps
n=6
n=12
n=18
n=24
Number of ingroup species included
B
Runtime of the phylogenomics steps of ERCnet (hours)
150
100
50
0
Dataset
n = 12, rep 1
n = 12, rep 2
n = 12, rep 3
9.49× speed increase
No parallelization
Parallelization optimized (24 threads)



Open a terminal

Edit a text file

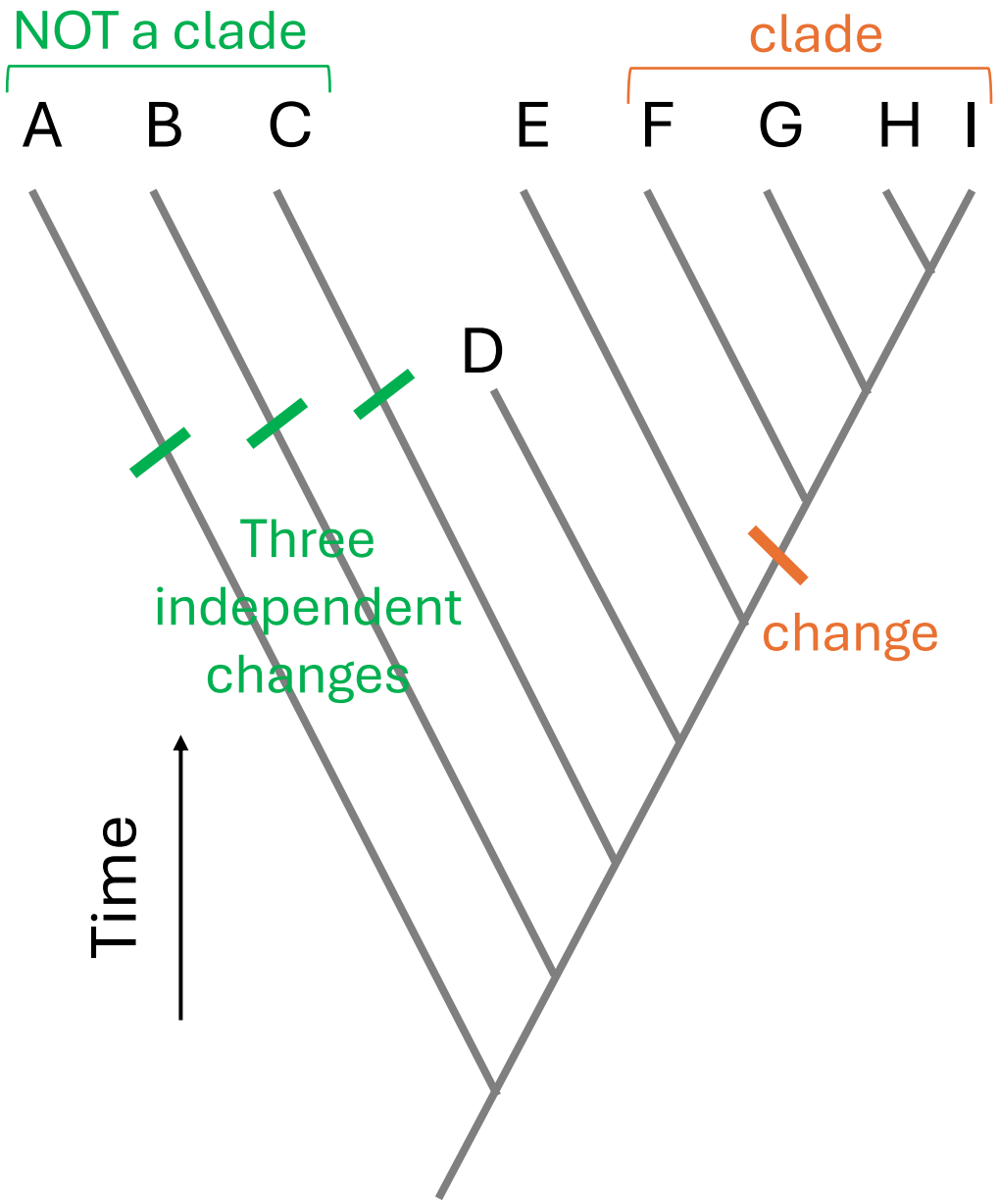




Enter info (e.g. ONID password, SSH commands, folder names, etc…)

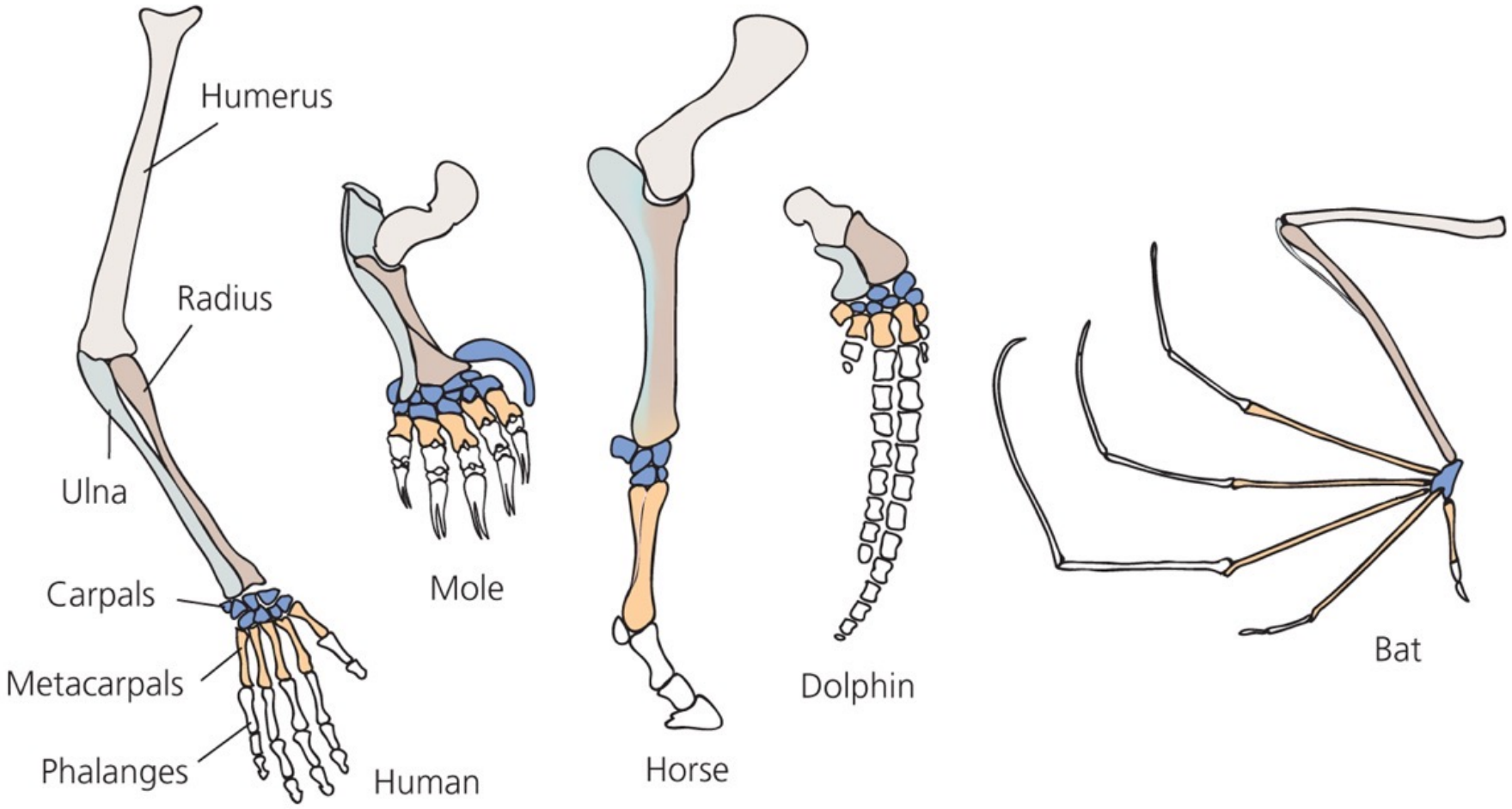
Connect to HPC via SSH plugin

Intro to phylogenetic trees



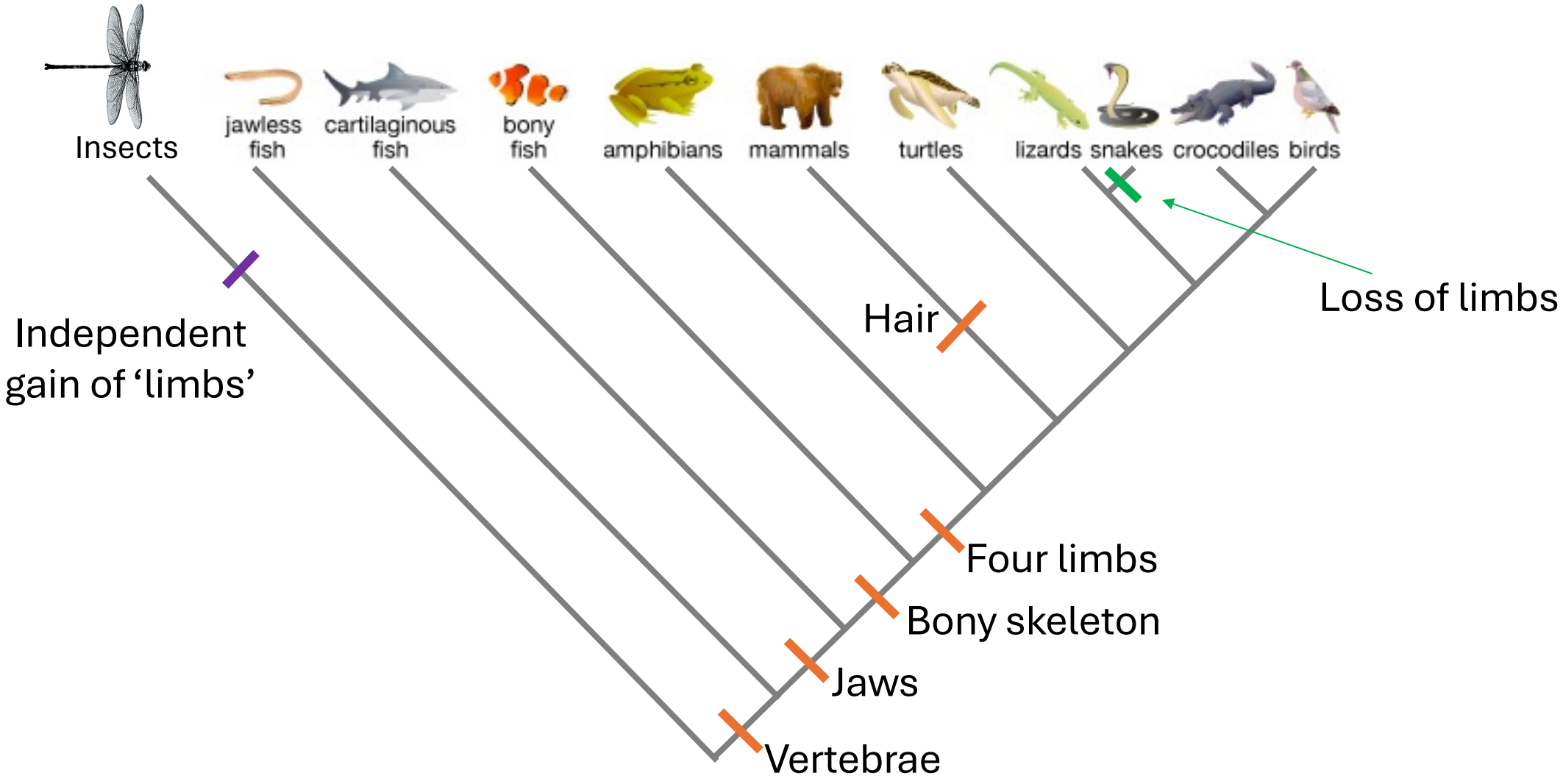
- The tips of the tree represent extant species
  - Extinct species are depicted by truncated branches (e.g. Species D)
- The nodes represent ancestral species that lived in the past
- Branches represent the passage of time
  - Evolutionary changes occur during the time between nodes
- A clade is a group of species that share a common ancestor
  - Clade = monophyletic

Homology



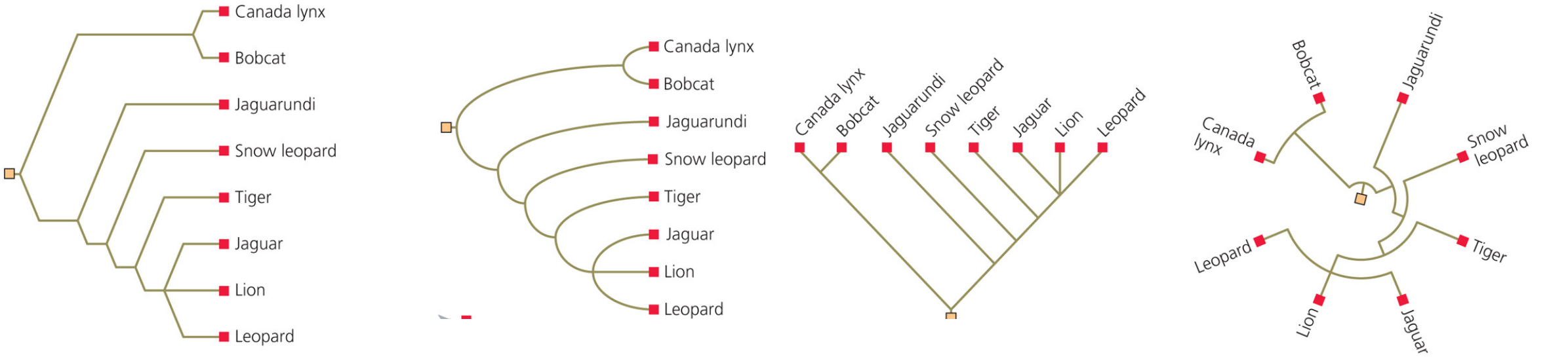
***Homology**: traits that are shared due to common ancestry*

‘Tree thinking’: framework for understanding when traits evolved



***Homology**: traits that are shared due to common ancestry*

Different ways of visualizing the exact same tree

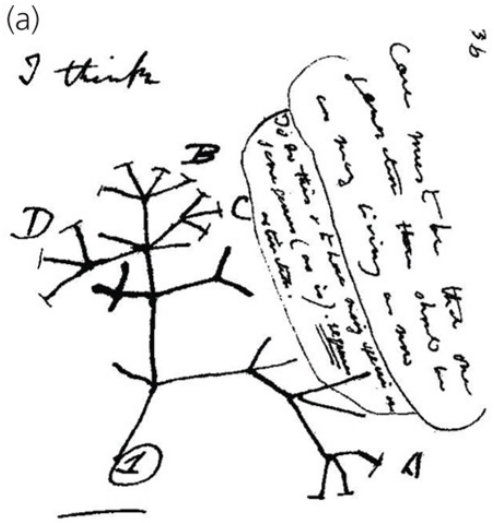


Newick (text format):

(Canada_lynx, Bobcat),(Jaguarundi,(Snow_leopard,(Tiger,(Jaguar, Lion, Leopard))

Binary format:

00101000 01000011 01100001 01101110 01100001 01100100 01100001 01011111 01101100 01111001 01101110 01111000 00101100 00100000
01000010 01101111 01100010 01100011 01100001 01110100 00101001 00101100 00101000 01001010 01100001 01100111 01110101 01100001
01110010 01110101 01101110 01100100 01101001 00101100 00101000 01010011 01101110 01101111 01110111 01011111 01101100 01100101
01101111 01110000 01100001 01110010 01100100 00101100 00101000 01010100 01101001 01100111 01100101 01110010 00101100 00101000
01001010 01100001 01100111 01110101 01100001 01110010 00101100 00100000 01001100 01101001 01101111 01101110 00101100 00100000
01001100 01100101 01101111 01110000 01100001 01110010 01100100 00101001 00101001



Phylogeny

- summary of a group's evolutionary history
- usually constructed from characters in living taxa
- useful for testing hypotheses about evolutionary history

Character

- Any feature that is scored for each taxon (morphological features, DNA base pairs, protein structure, etc.)
- They should be independent and homologous.
- Can be either ancestral or derived
  - Ancestral – retained from common ancestor
  - Derived – newly-evolved character

Synapomorphy: a shared, derived character

- shared = 2+ taxa have one character state
- derived = novel with respect to ancestral state
- Synapomorphies define clades

Homoplasy: "noise" in the data due to convergent evolution and reversals

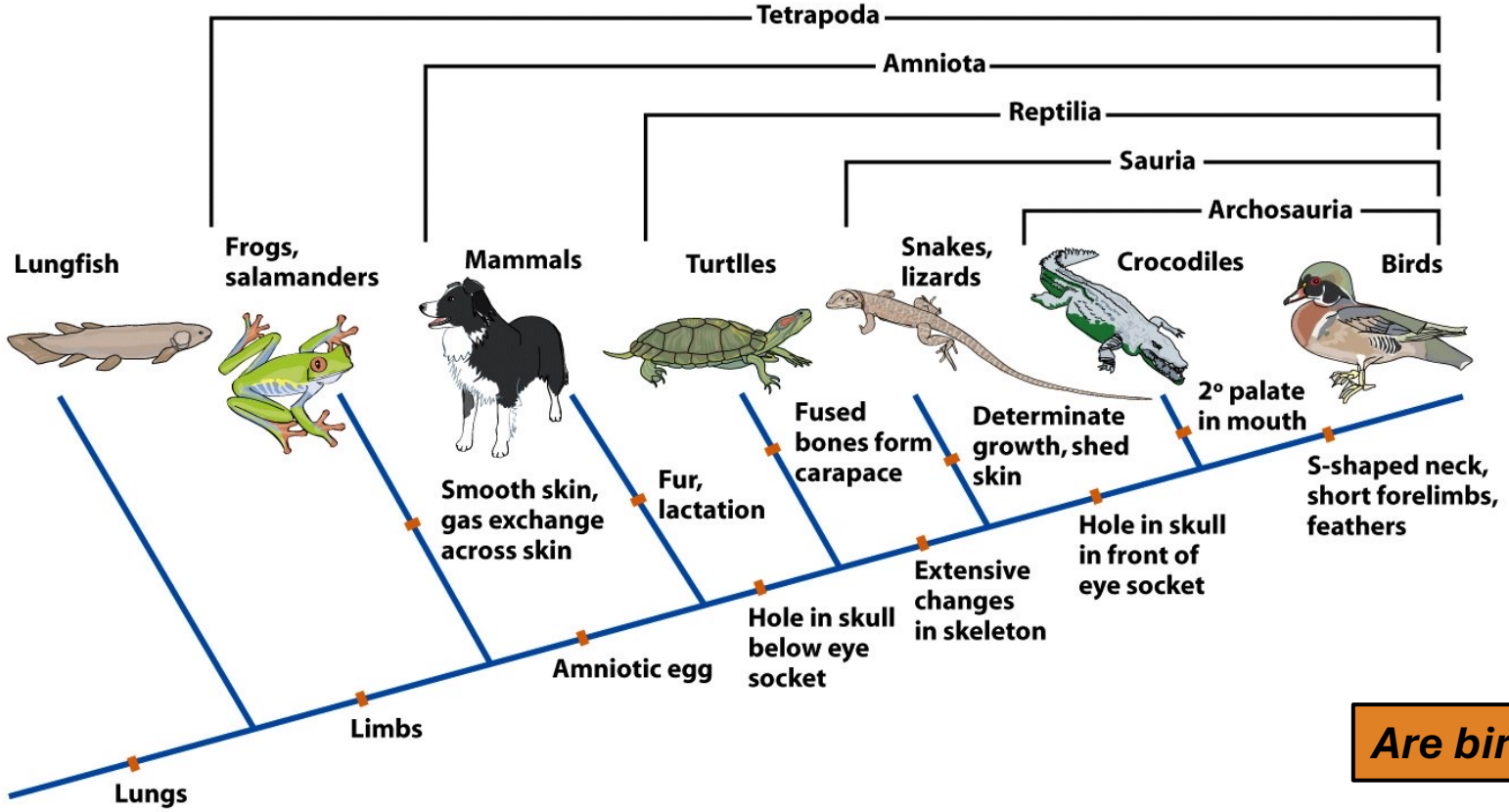
- complicates phylogenetic reconstruction
- common in DNA sequence data (only four states: A,T,C,G)
- Solution: use many informative characters

Parsimony:

- Logical argument that the simplest explanation is the best explanation
- Simplest = fewest transitions

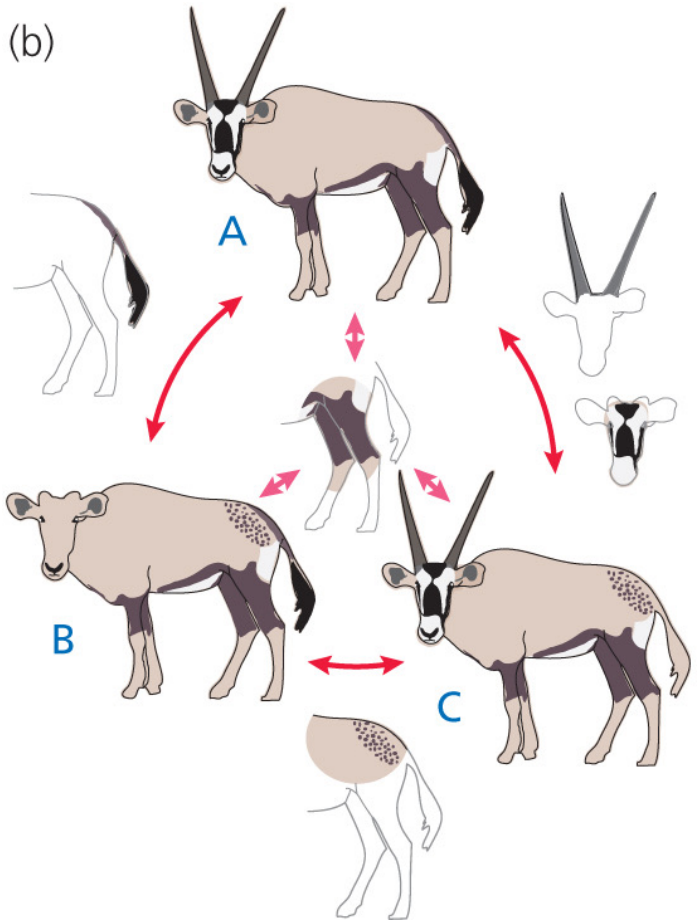
Examples of synapomorphies

Synapomorphy	Clade that it defines
feathers	Birds
hole below eye socket	Reptilia

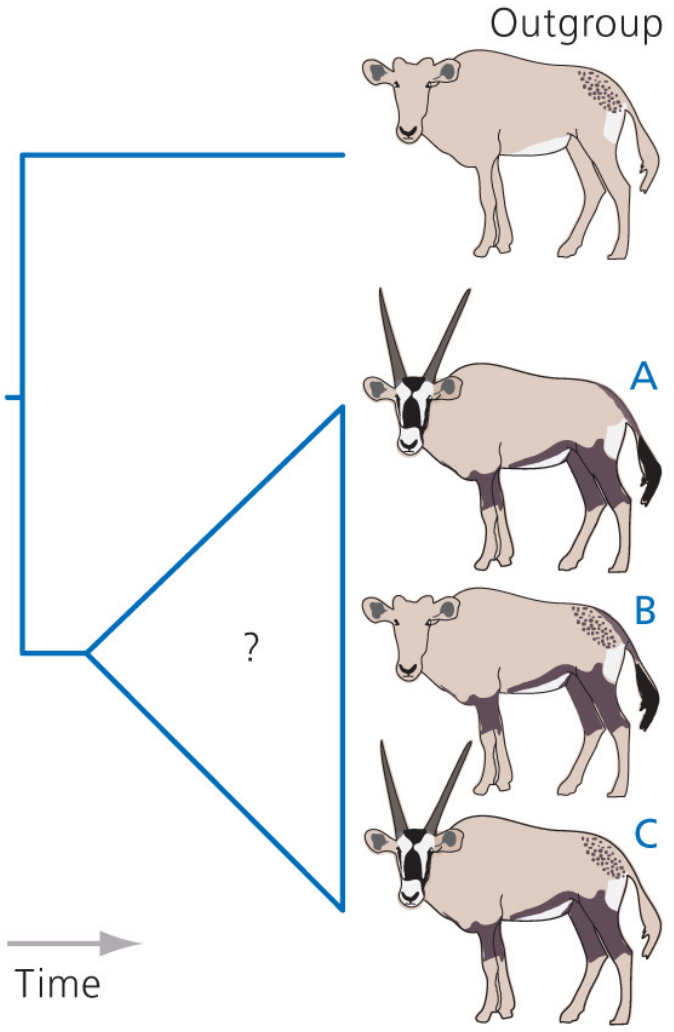


Are birds reptiles?

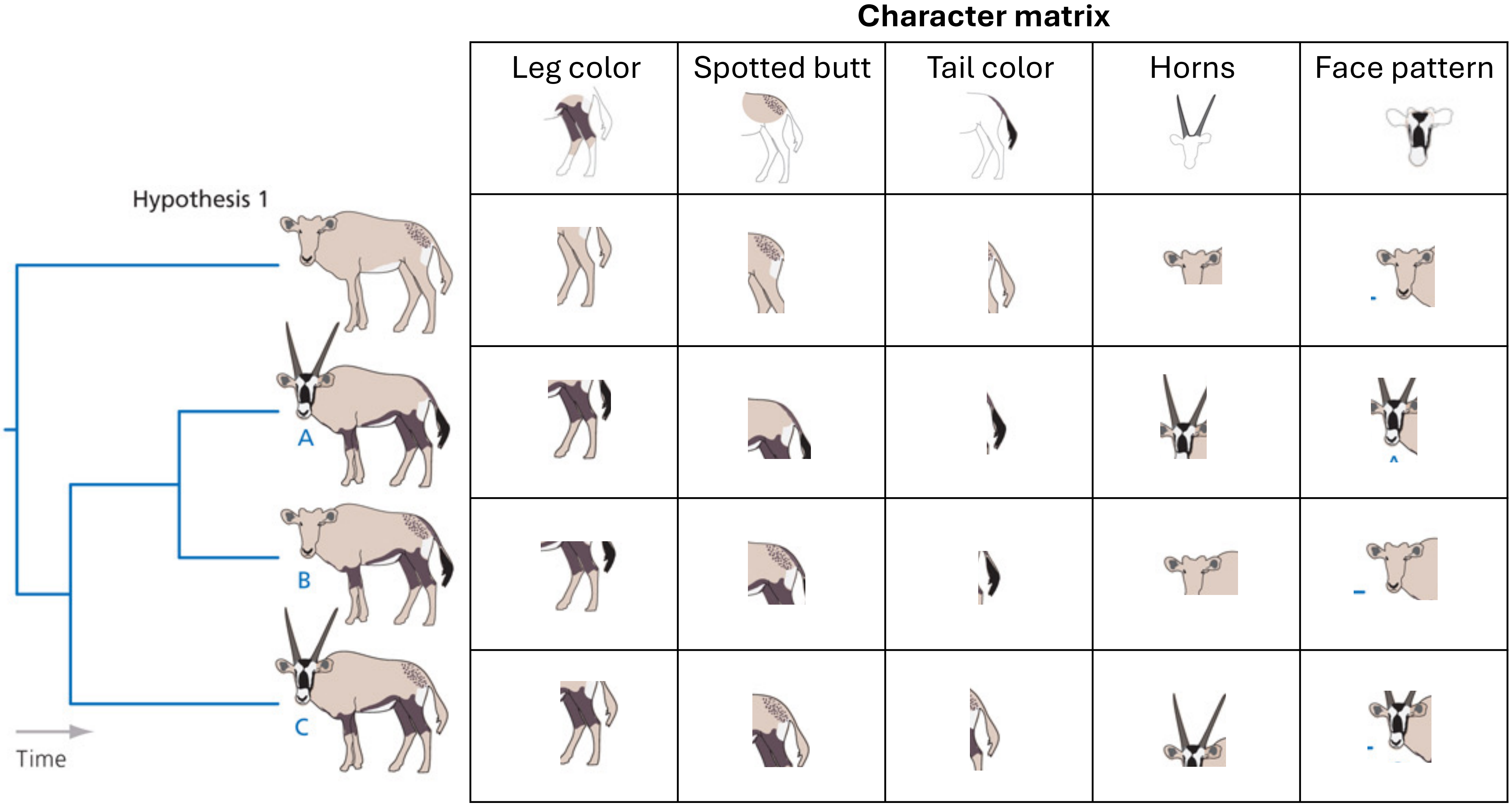
Extant species



We usually don't know which traits are derived and ancestral

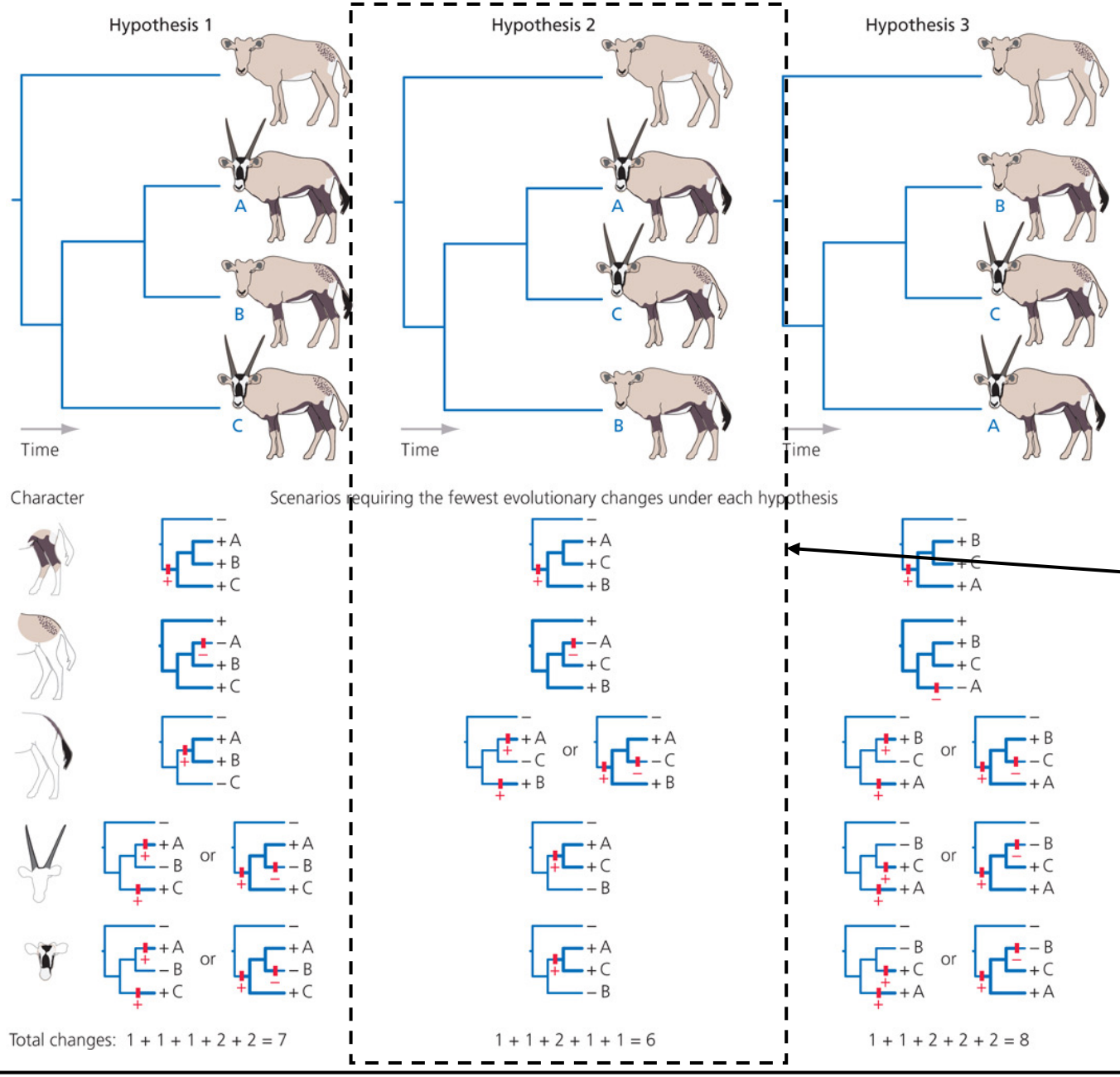


Calculating a parsimony score for a tree based on characters



Parsimony score	1	1	1	2	2	Total: 7

Parsimony-based tree search

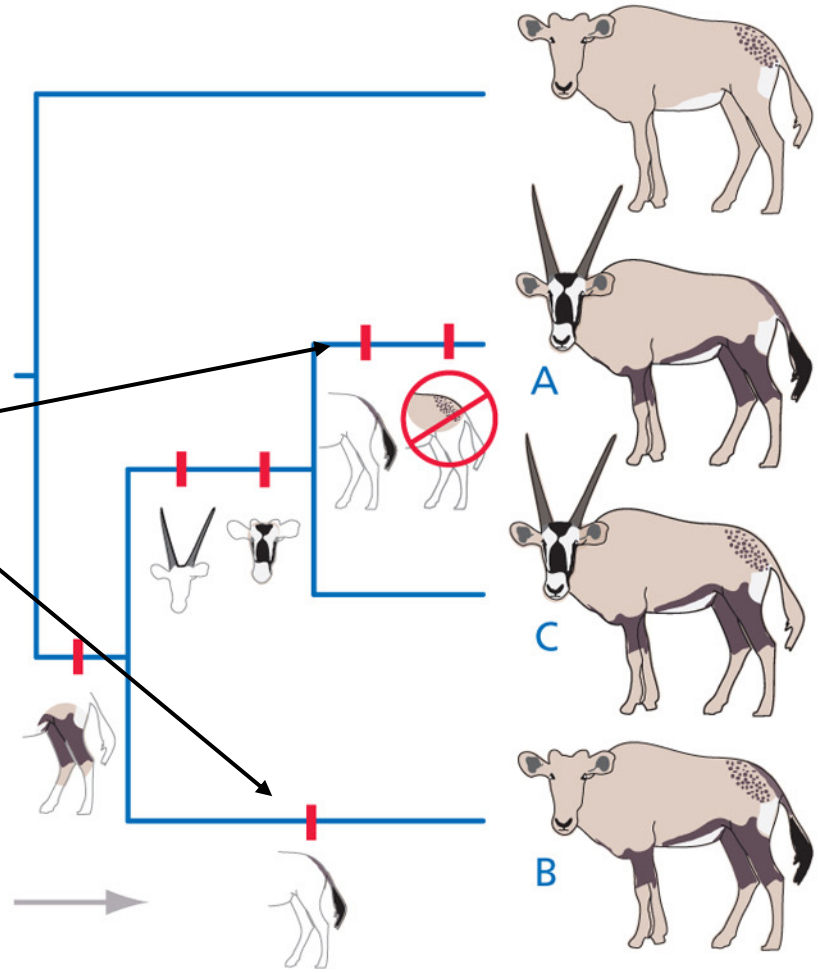


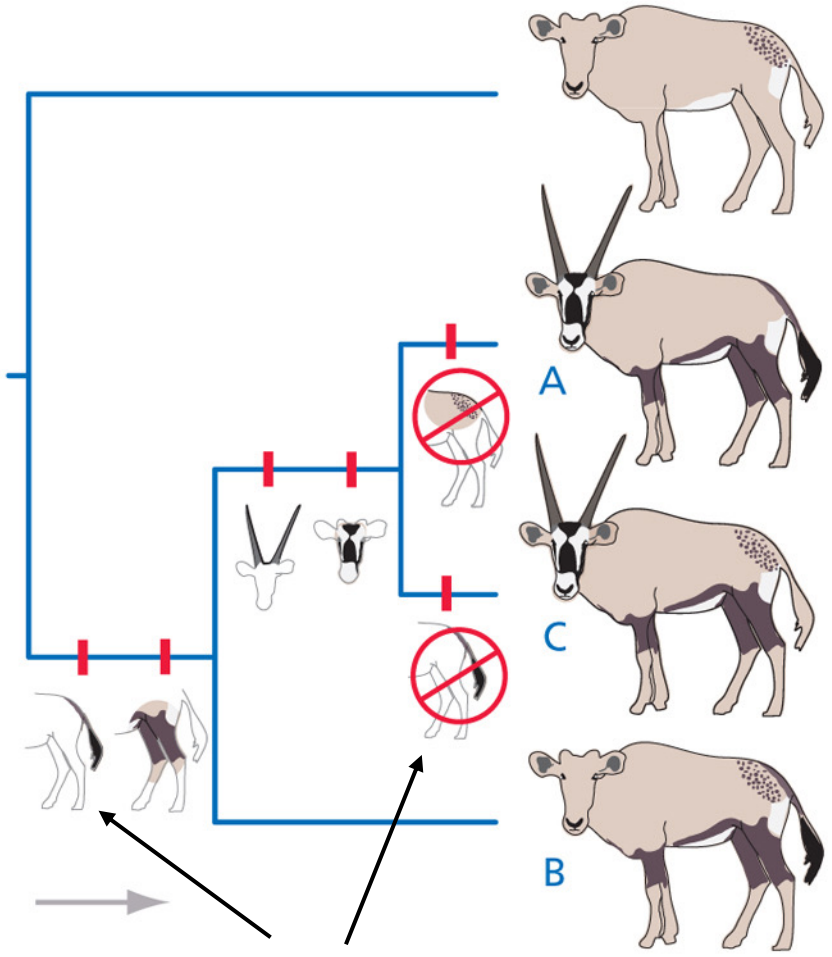
The most parsimonious tree is favored

- Most parsimonious = fewest transitions

Two equally parsimonious histories

The same transition occurred in two independent lineages (convergent evolution)



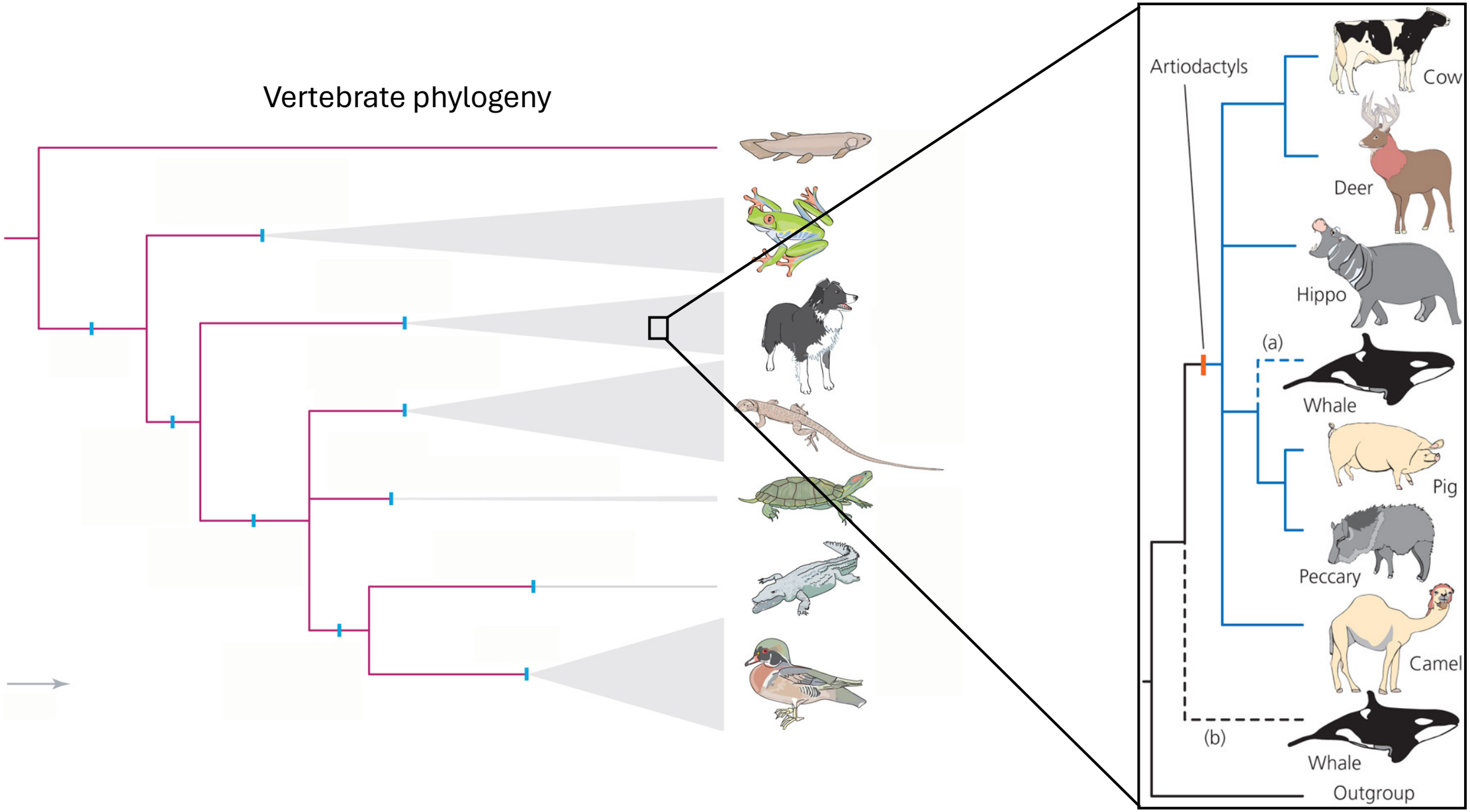


A back-transition occurred

Homoplasy:

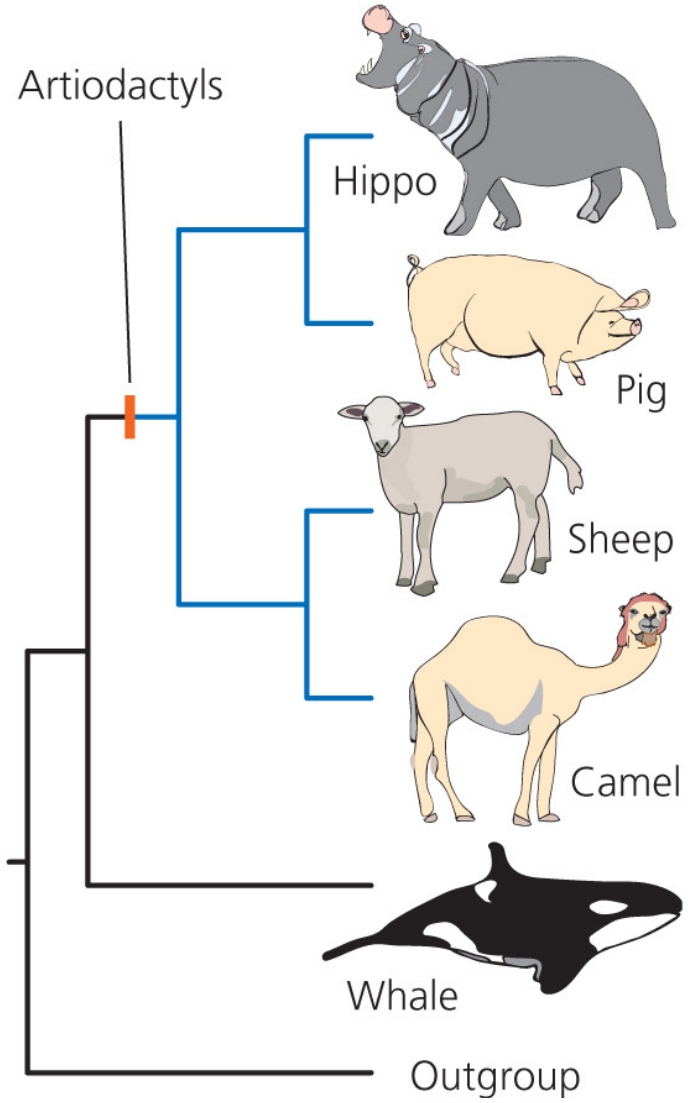
- When characters experienced convergence or back-transitions
- Creates major problems for phylogenetic inference (especially with parsimony)

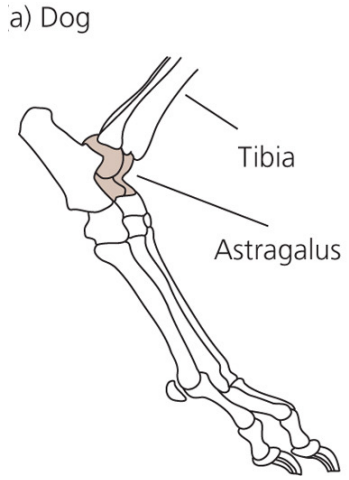
What are whales???



Parsimony phylogenetic analysis using morphological traits

Topology favored by parsimony-based analysis of morphological characters

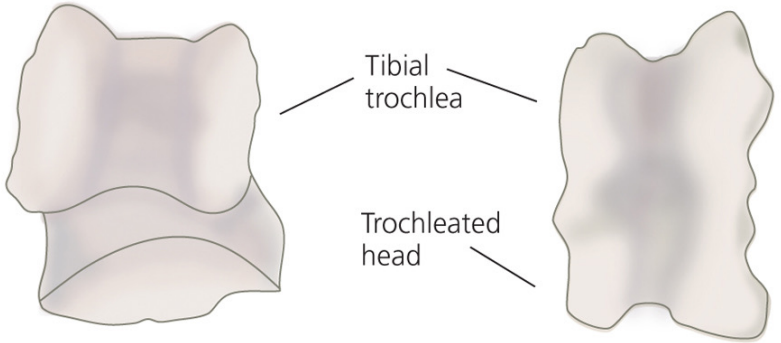




- Bones in the hind leg are traditionally a useful character for defining the Artiodactyl group

(b) Tapir—a nonartiodactyl ungulate

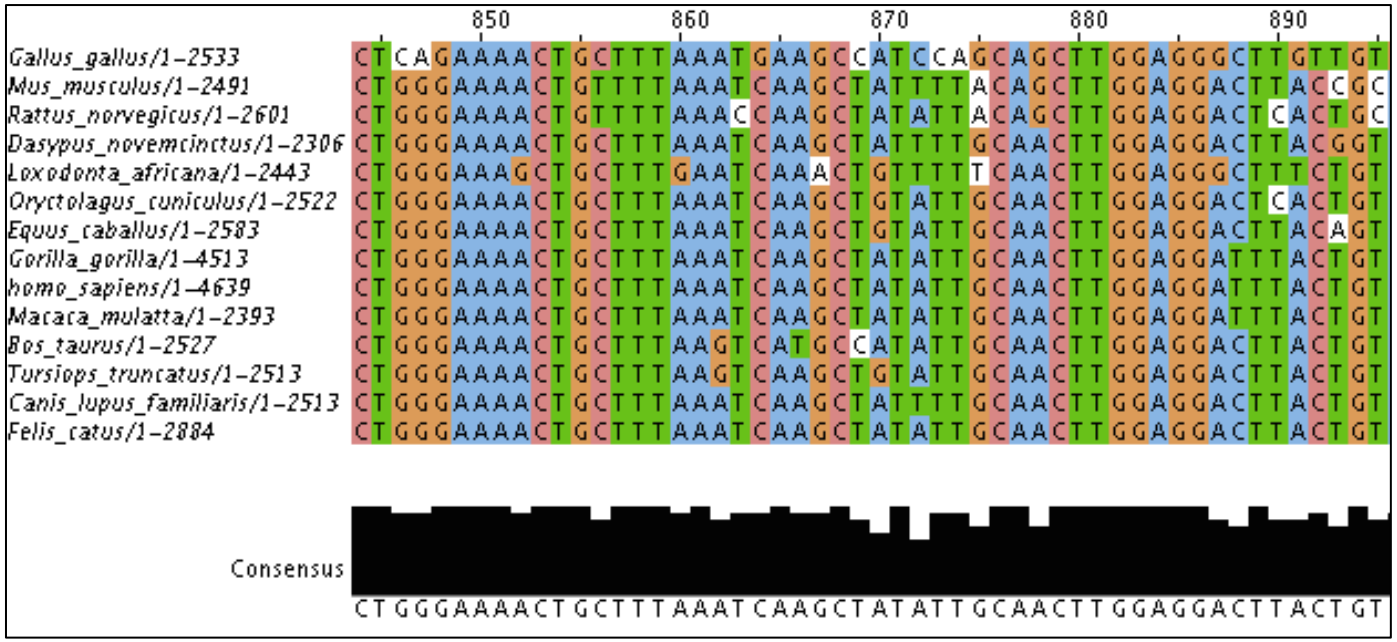
(c) Deer—an artiodactyl ungulate



Molecular homology

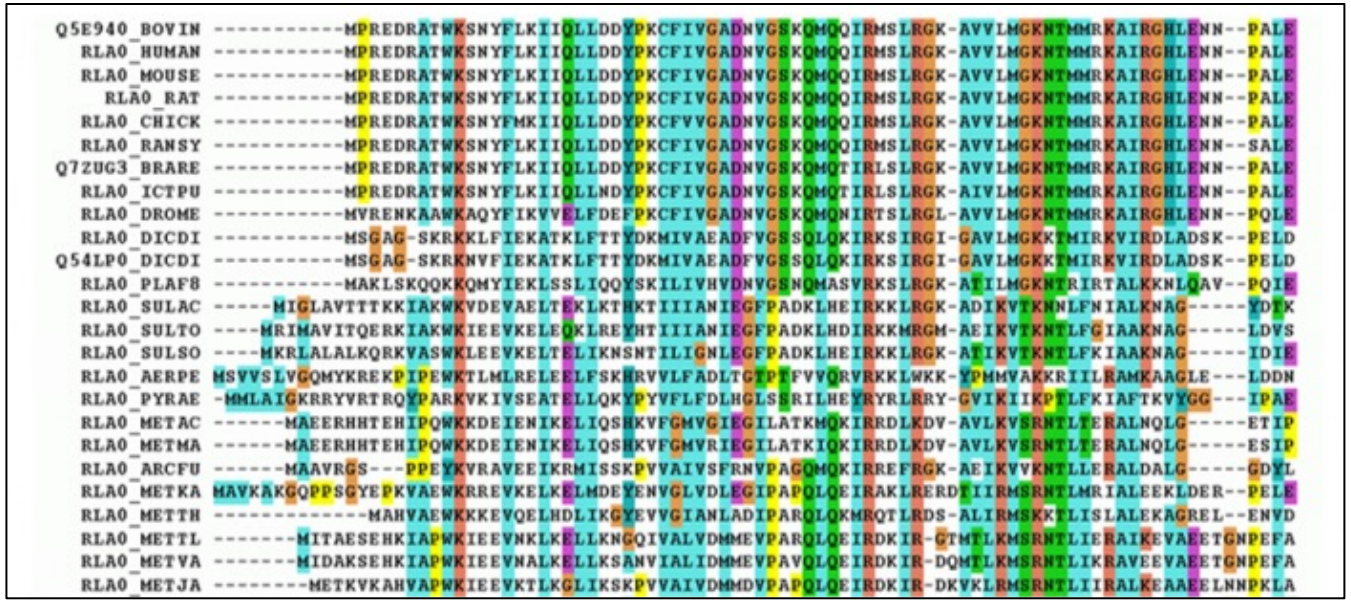
- The DNA sequences are *nearly* identical from these four different species
  - This reflects the shared ancestry of the species and that *Gene A* is a homologous gene shared by these four species
- The differences in the DNA sequences are due to **mutations.**
  - Species 1 and 2 share two mutations, meaning the mutation occurred in their common ancestor but not in the ancestor of Species 3 and 4

Molecular (DNA or Amino Acid sequence) phylogenetics



DNA sequences can be used to make a character matrix

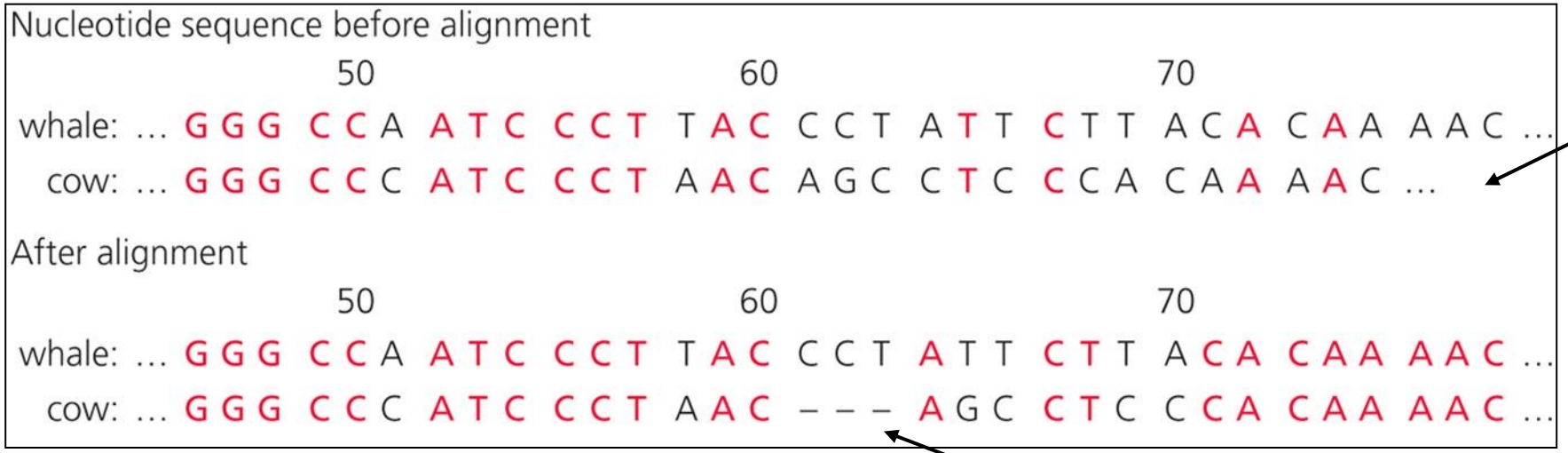
- DNA samples are sequenced from multiple different species
- Sequences are 'aligned' so that each column (i.e. site) in the alignment represents a homologous character

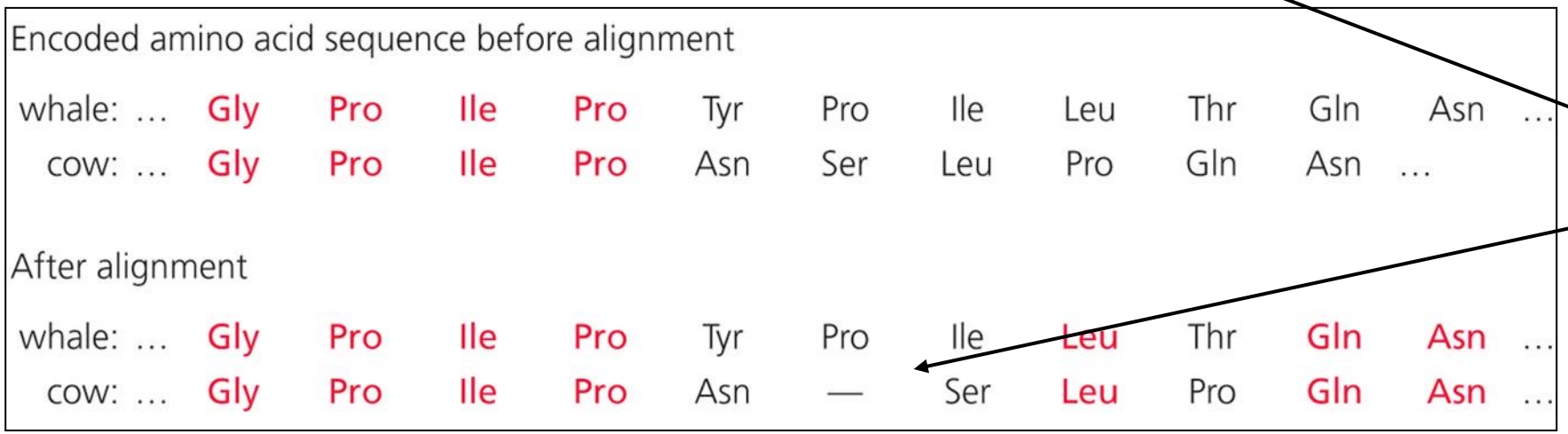


Protein (amino acid) sequences can be used to make a character matrix

- Protein's can be 'sequenced' with mass spectrometry (rare)
- Protein sequence can be inferred from DNA sequence once it's been sequenced (common)

Multiple sequence alignment (MSA)





Sequences are different lengths

Multiple sequence alignment:

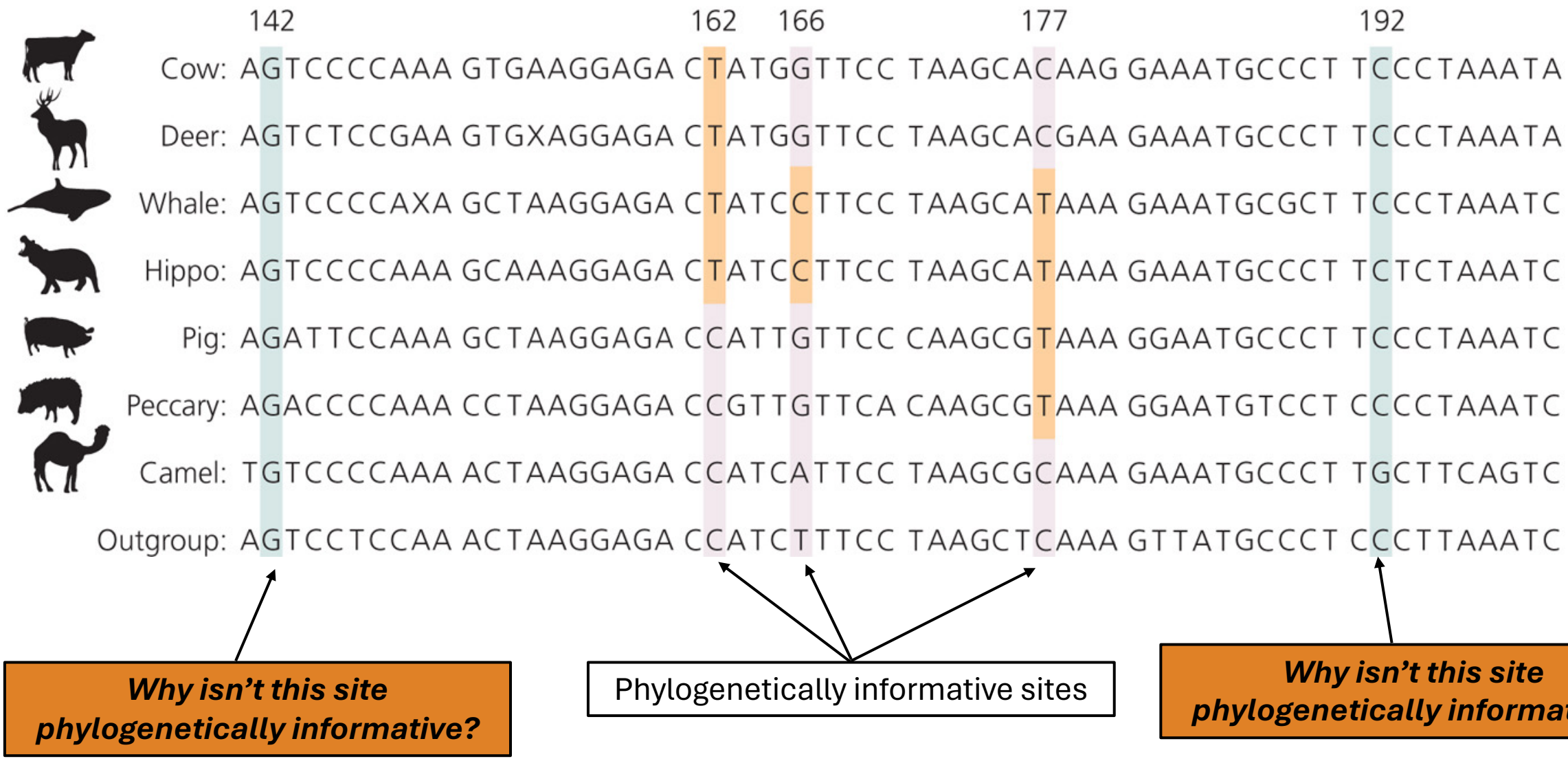
- 'lining up' sequences so that columns are most likely to represent homologous sites
- Often involves introducing gaps

Indel

- Insertion or deletion

How could we learn whether it's an insertion or deletion?

Using DNA alignments to perform phylogenetic inference



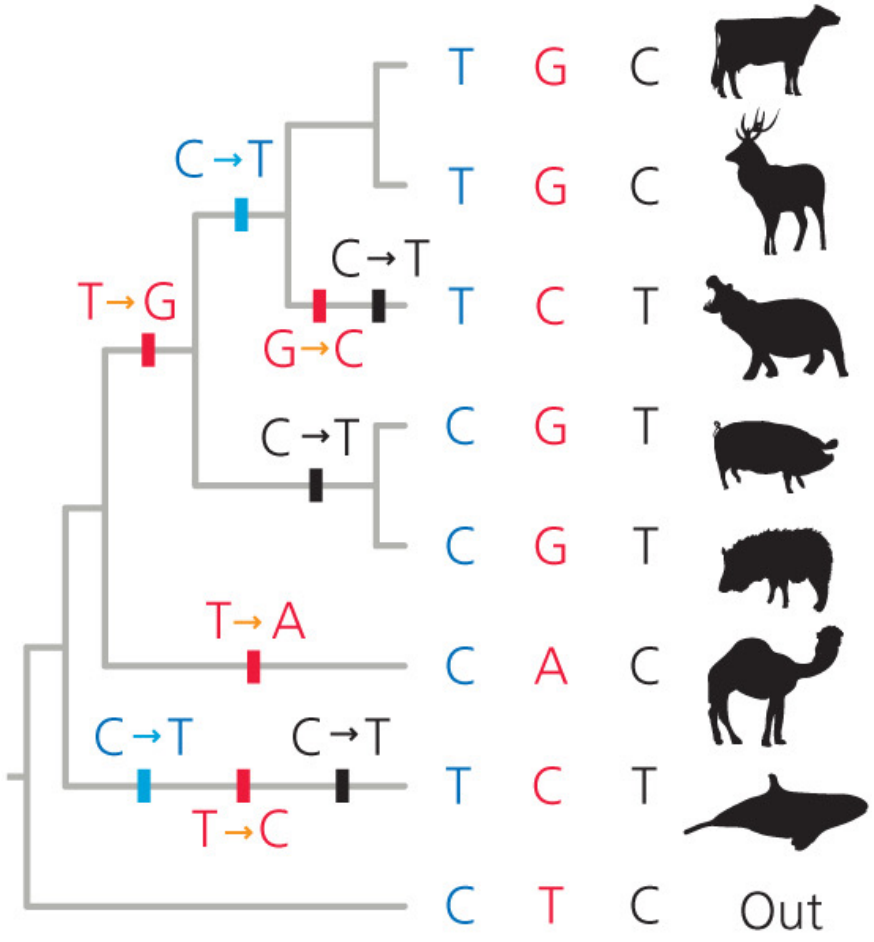
Why isn't this site phylogenetically informative?

Phylogenetically informative sites

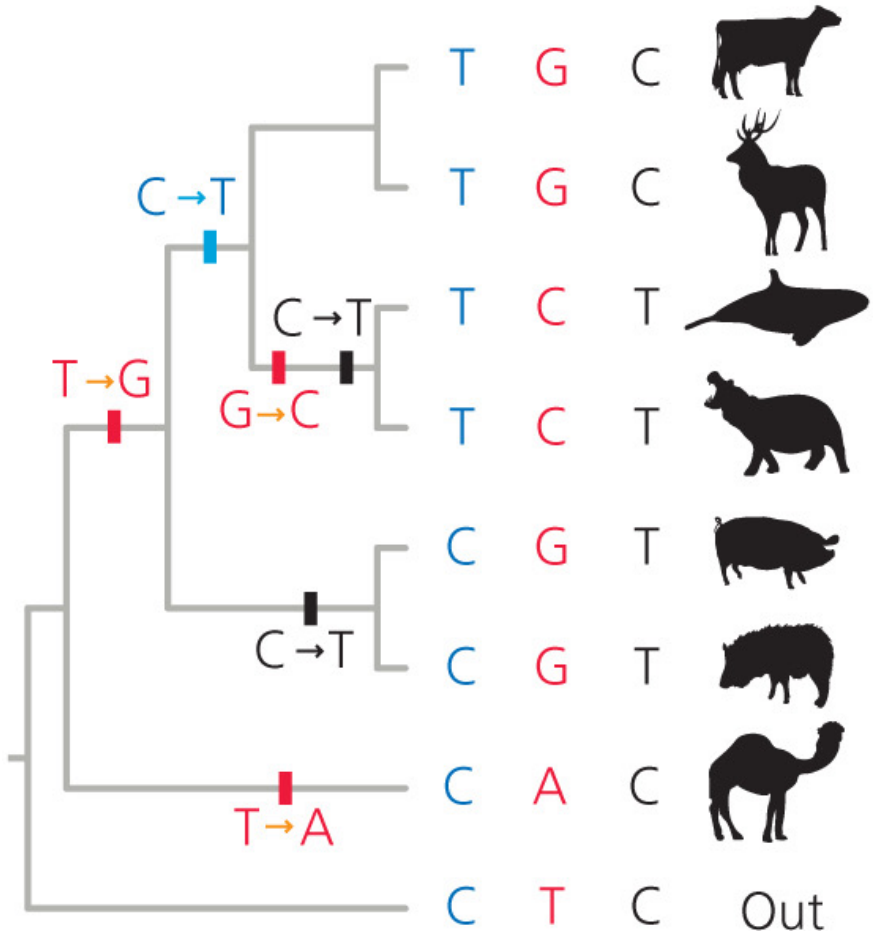
Why isn't this site phylogenetically informative?

Using DNA alignments to perform phylogenetic inference

Hypothesis 1: whales are sister to all other artiodactyls



Hypothesis 2: whales are sister hippos (within artiodactyls)



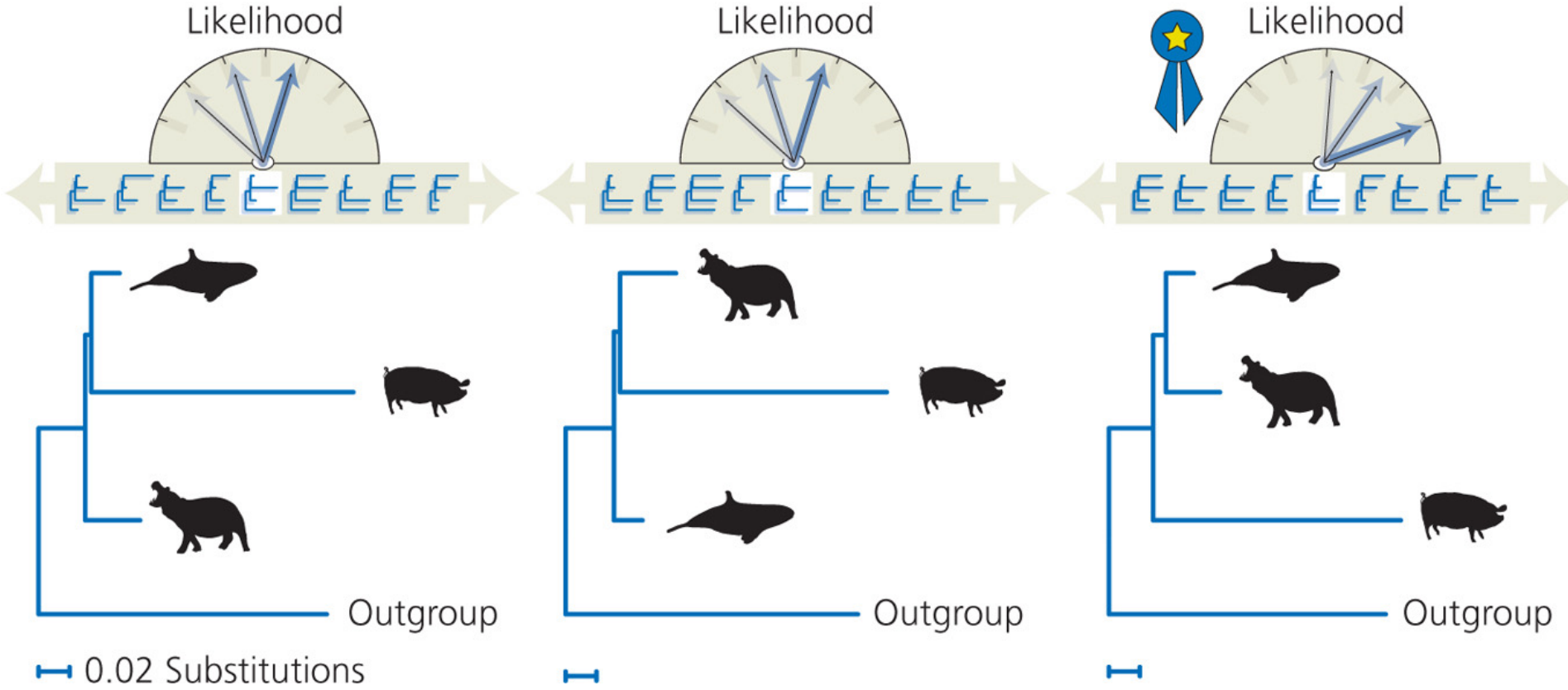
Which topology is favored by parsimony analysis of these three phylogenetical informative sites?

Is there homoplasy at these sites?

Maximum likelihood (ML) phylogenetic inference

Likelihood score:

- A statistic that quantifies the probability of the observed alignment, given a particular tree



Maximum likelihood phylogenetic inference:

- Calculate the likelihood score for many trees and choose the one with the highest (i.e. maximum) likelihood score

Maximum likelihood (ML) phylogenetic inference

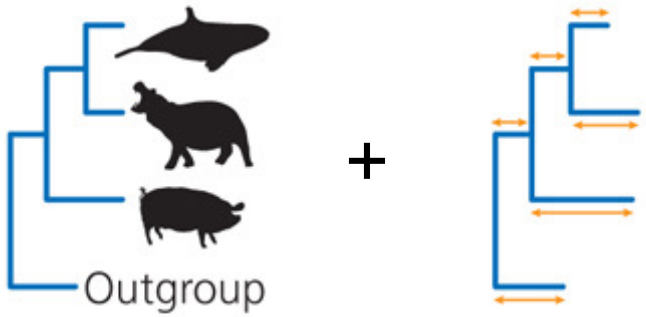
The user provides multiple sequence alignment:

- MSAs are the input data for ML inference

The data

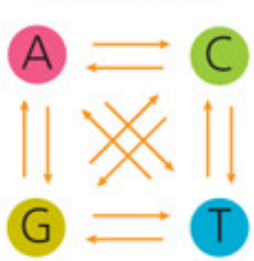
The algorithm choses a random tree

- Topology and branch lengths



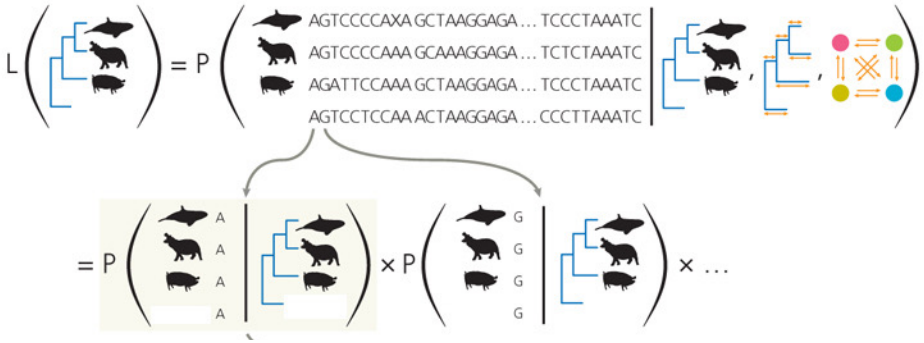
ML uses models of DNA sequence evolution:

- a set of numbers giving the probability of each possible substitution
- Models can account for homoplasy!



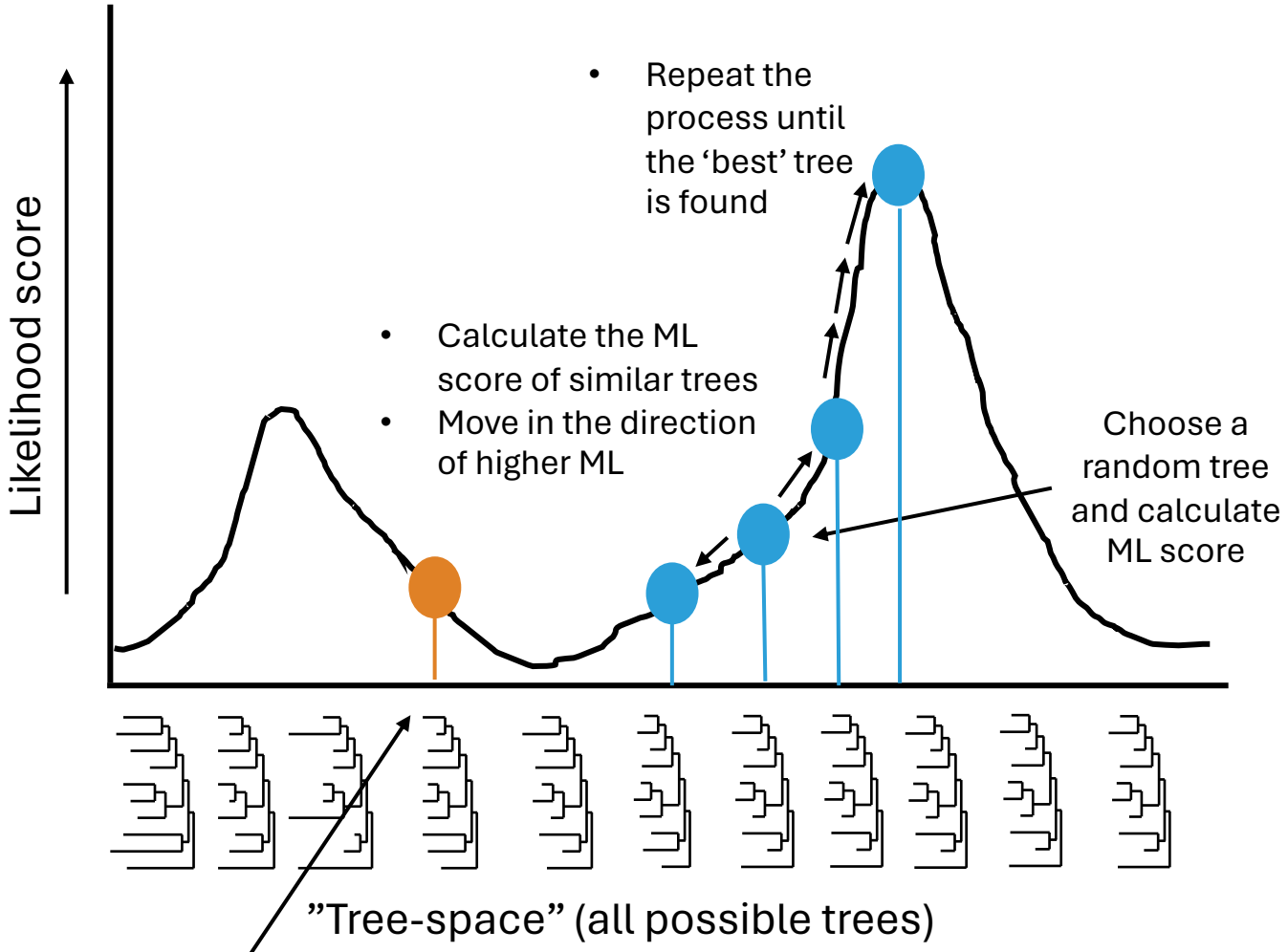
Calculating the likelihood score for a tree

Likelihood score of a tree = Probability of the observed alignment given the tree and the model of evolution



Probability of site 1 (given the tree) X Probability of site 2 (given the tree) = Likelihood score

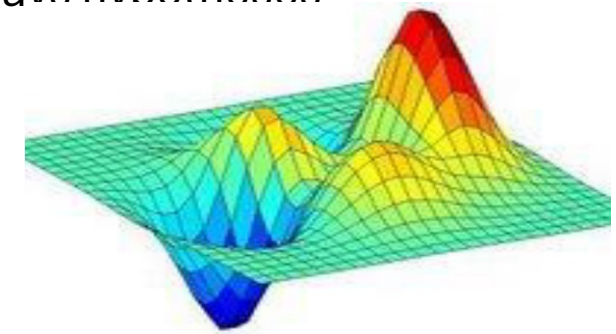
Maximum likelihood (ML) phylogenetic inference



Would the algorithm select the 'globally optimal tree' if it randomly started here?

Maximum likelihood inference:

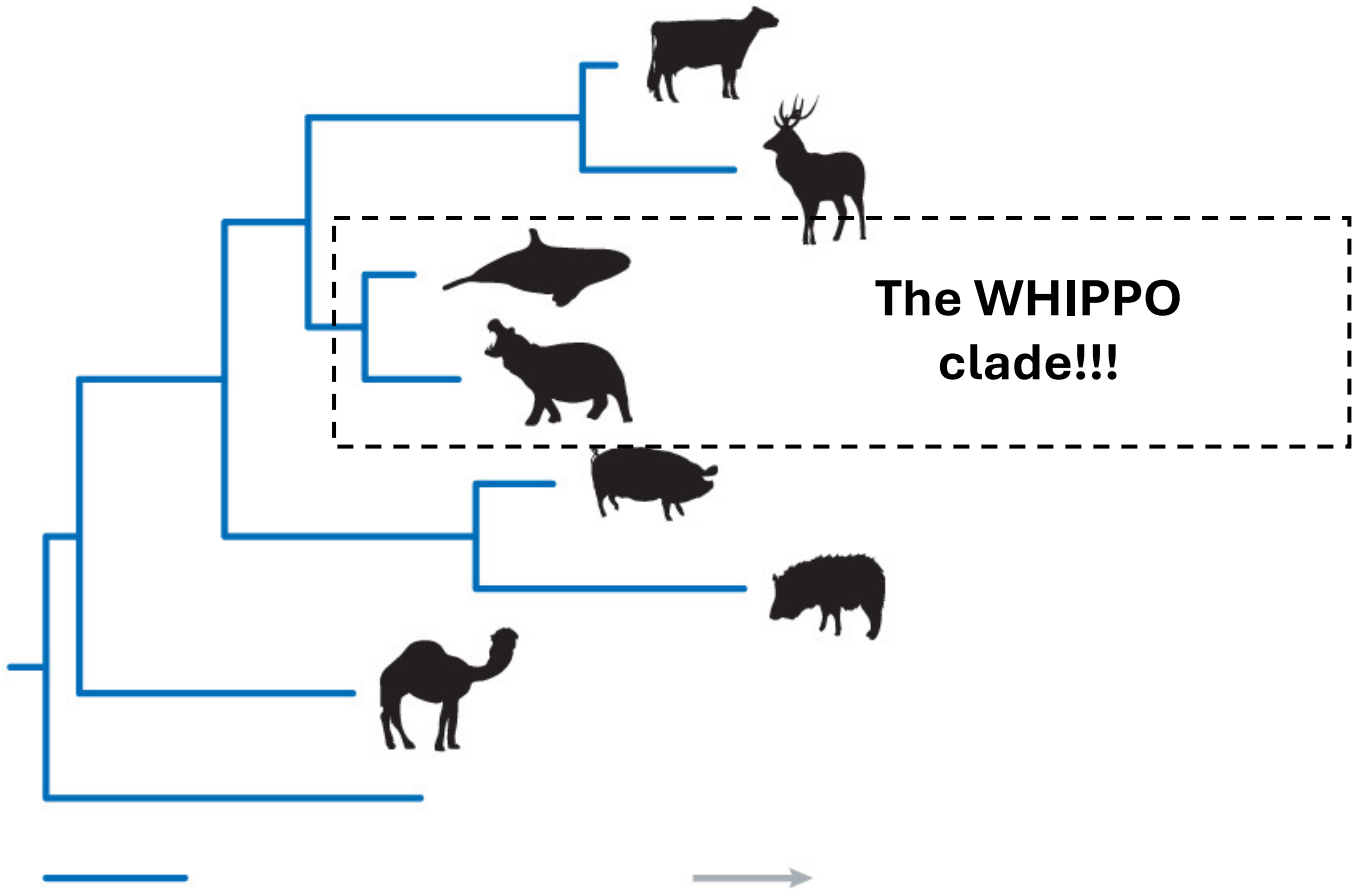
- Sometime called 'tree search'
- Iterative (occurs through multiple repeated steps)
- 'Guess', 'check', 'repeat'
- The number of possible trees is usually LARGE
- Requires powerful computers to search many possible trees
- Even with large computers, it's impossible to test every tree
  - This is one of the reasons trees are always hypotheses



Tree-space represented in 3-dimensions

ML trees support the Whippo clade

The current leading hypothesis

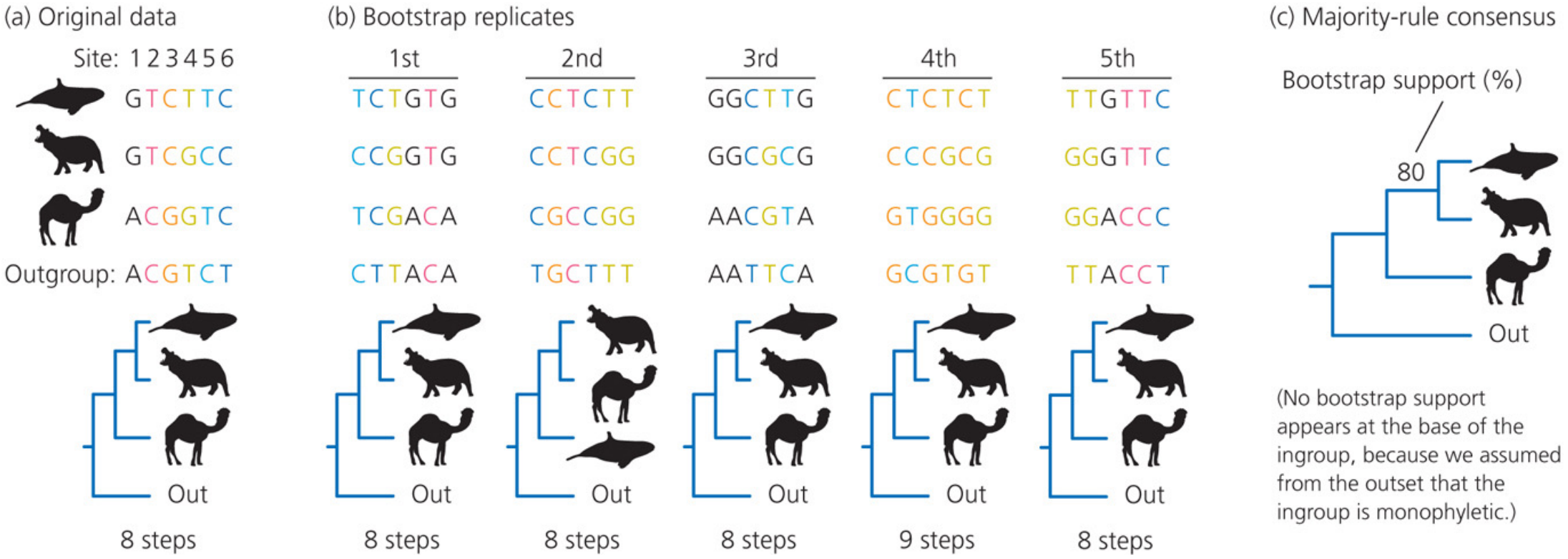


0.07 (substitutions per site in alignment)

Notes about interpreting phylogenies:

- Phylogenies ALWAYS represent hypotheses
- We will never know the answer
- We can only use phylogenetics to favor some hypotheses over others (not prove or disprove)

Assessing our confidence in our chosen tree with bootstrap resampling of the original data

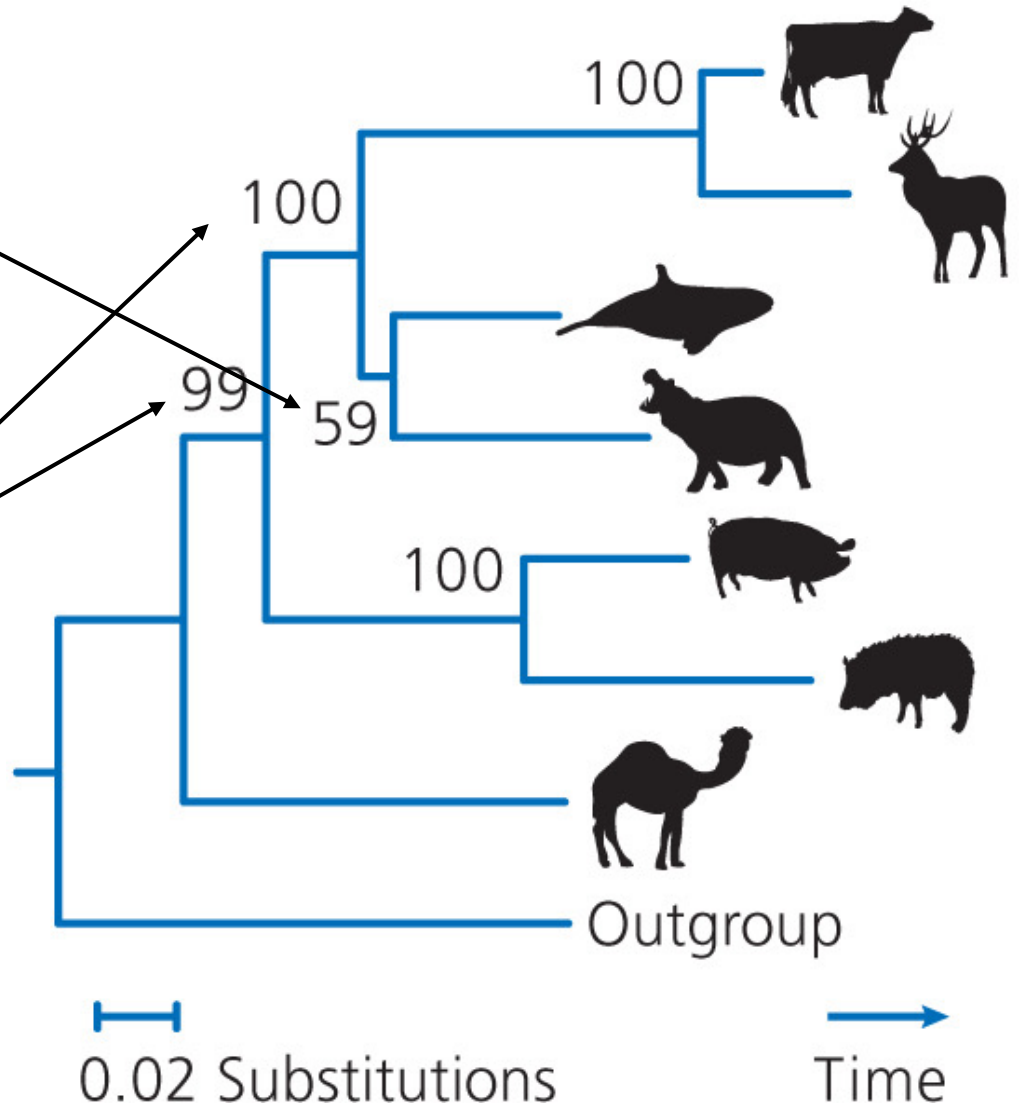


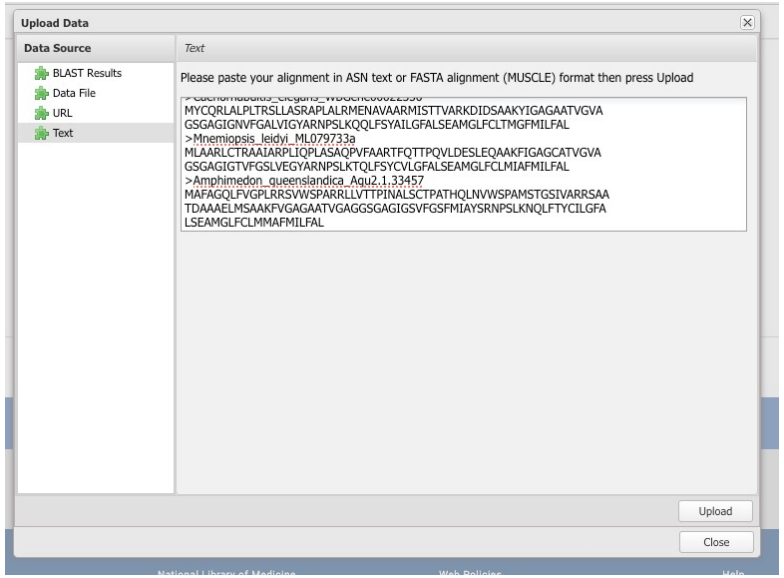
How is a bootstrap support score similar to a p-value? How is it different?

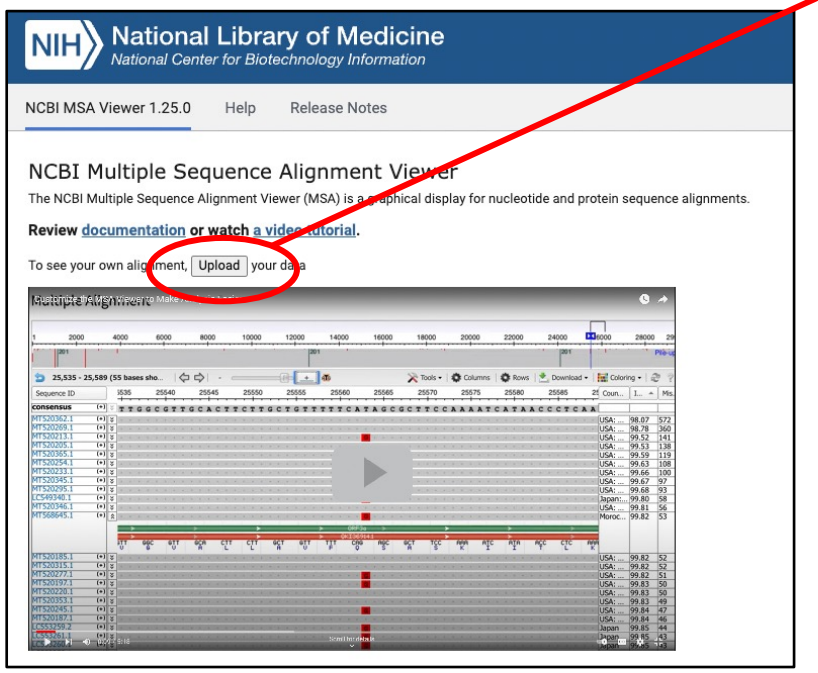
Assessing our confidence in our chosen tree using bootstrap resampling of the original data

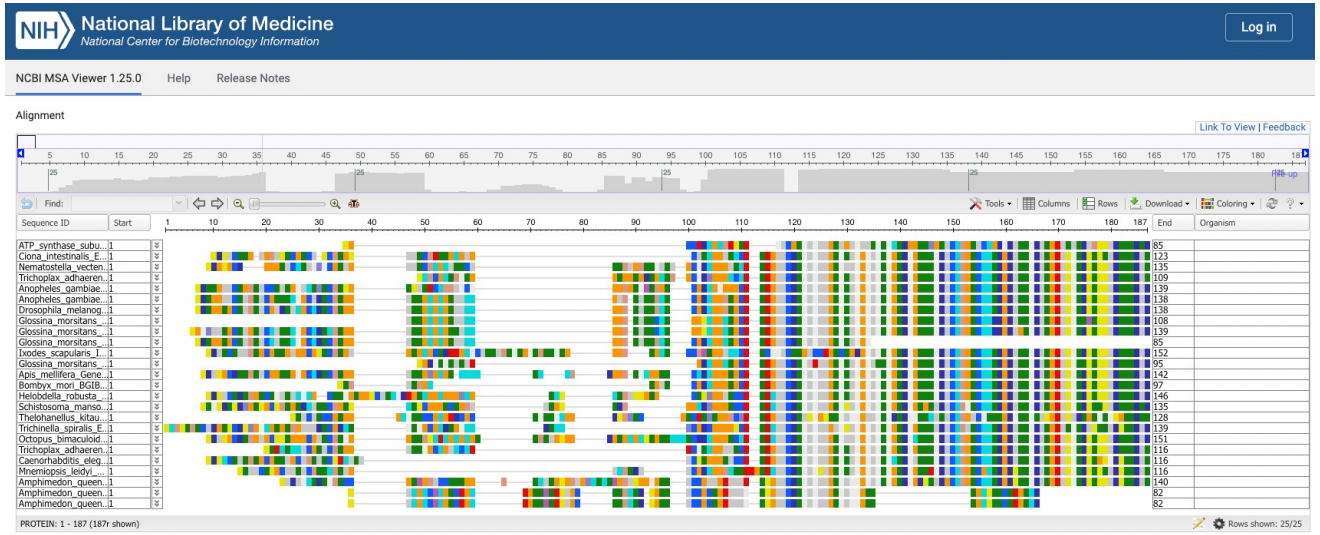
The ‘whippo’ clade is actually not very well- supported

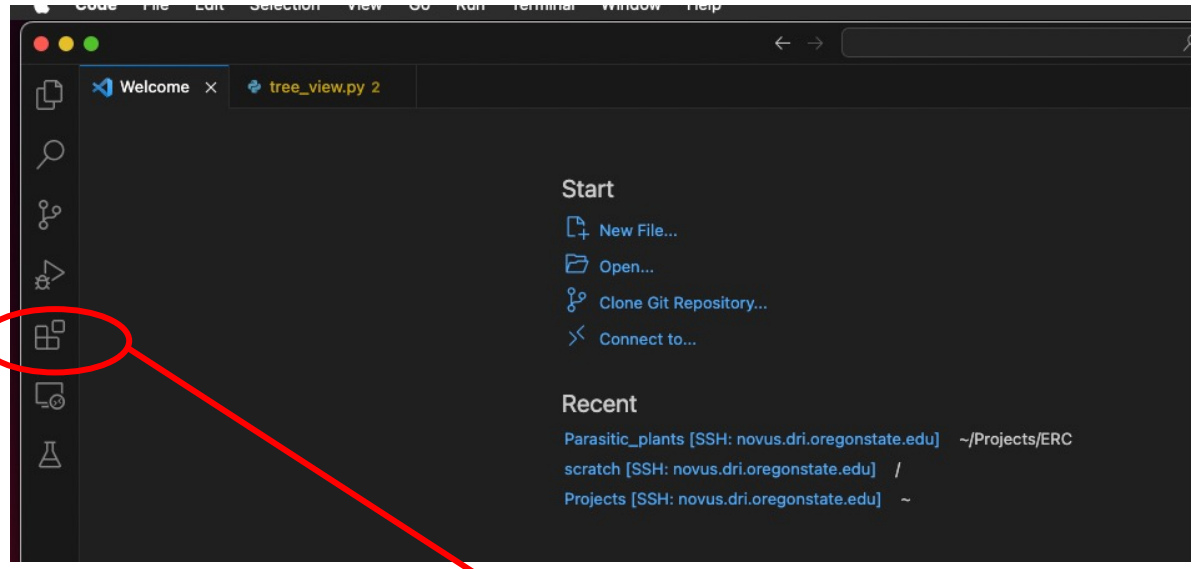
However, the clade that places whales as an artiodactyl is well-supported

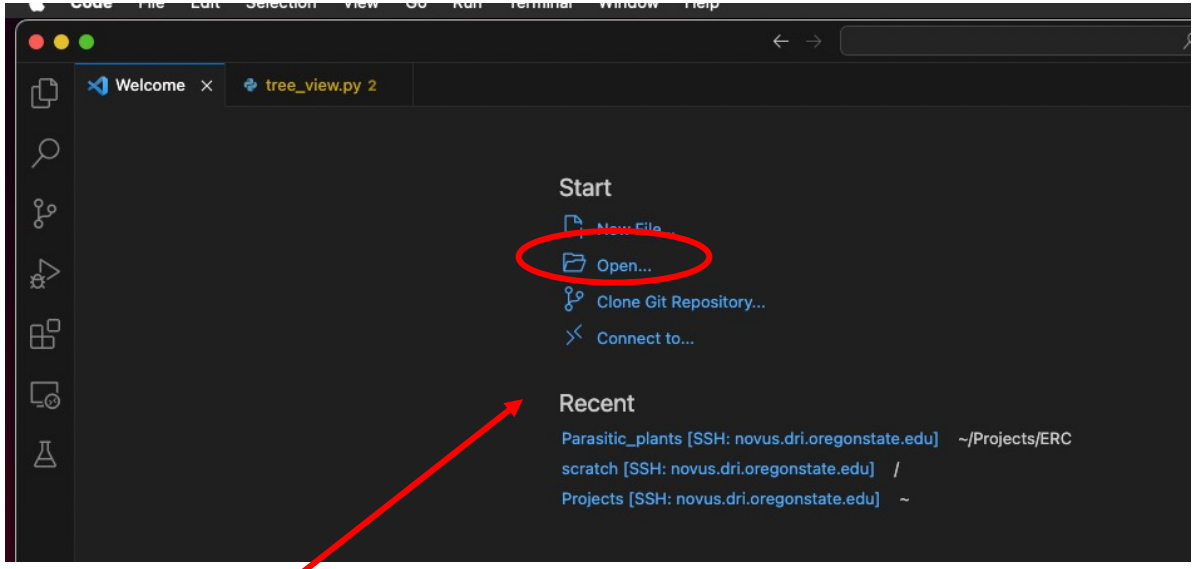


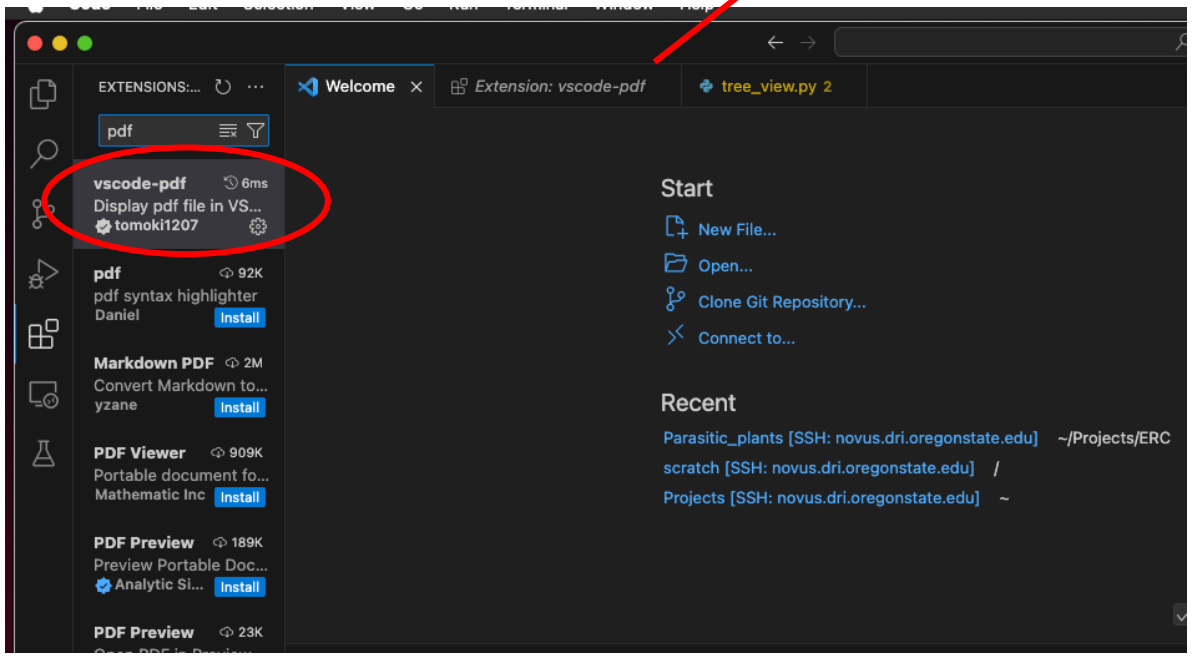

Upload Data
Data Source
BLAST Results
Data File
URL
Text
Text
Please paste your alignment in ASN text or FASTA alignment (MUSCLE) format then press Upload
MYCQRLALPLTRSLLASRAPLALRMENAVAARMISTTVARKDIDSAAKYIGAGAATVGVA
GSGAGIGNVFGALVIGYARNPSLKQQLFSYAILGFALSEAMGLFCLTMGFMILFAL
>Mnemiopsis_leidyi_ML079733a
MLAARLCTRAAIARPLIQPLASAQPVFAARTFQTTPQVLDESLEQAAKFIGAGCATVGVA
GSGAGIGTVFGSLVEGYARNPSLKTQLFSYCVLGFALSEAMGLFCLMIAFMILFAL
>Amphimedon_queenslandica_Aqu2.1.33457
MAFAGQLFVGPLRRSVWSPARRLLVTTPINALSCTPATHQLNVWSPAMSTGSIVARRSAA
TDAAAELMSAAKFVGAGAATVGAGGSGAGIGSVFGSFMIAYSRNPSLKNQLFTYCILGFA
LSEAMGLFCLMMAFMILFAL
Upload
Close

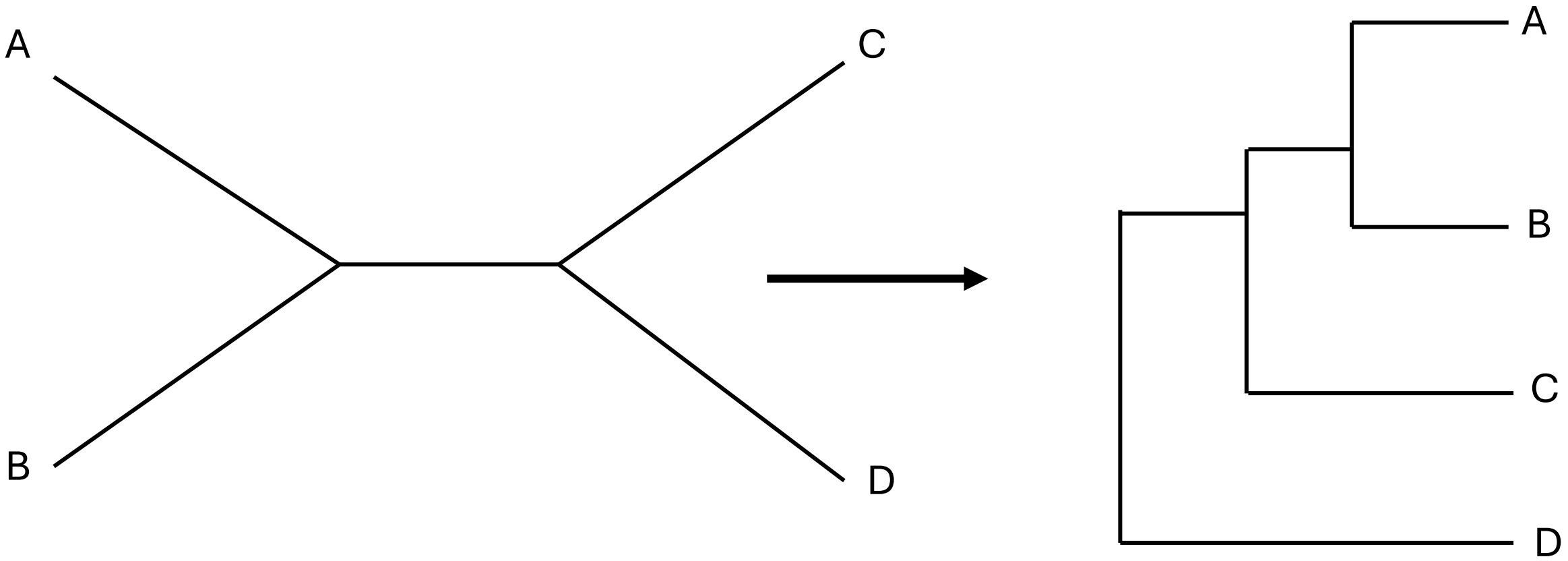

NIH
National Library of Medicine
National Center for Biotechnology Information
NCBI MSA Viewer 1.25.0
Help
Release Notes
NCBI Multiple Sequence Alignment Viewer
The NCBI Multiple Sequence Alignment Viewer (MSA) is a graphical display for nucleotide and protein sequence alignments.
Review documentation or watch a video tutorial.
To see your own alignment, Upload your data

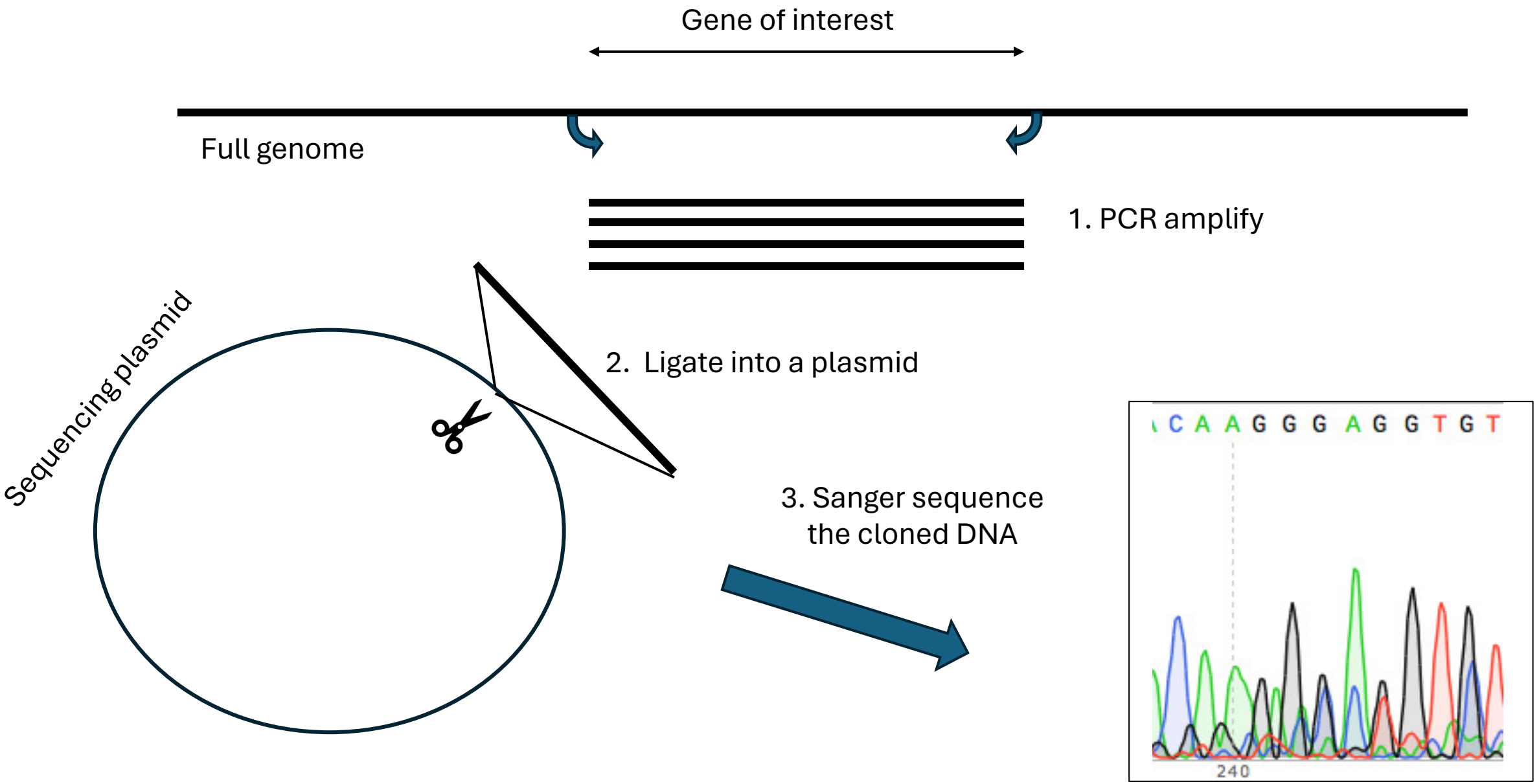

NIH
National Library of Medicine
National Center for Biotechnology Information
Log in
NCBI MSA Viewer 1.25.0
Help
Release Notes
Alignment
Link To View | Feedback


Welcome
tree_view.py 2
Start
New File...
Open...
Clone Git Repository...
Connect to...
Recent
Parasitic_plants [SSH: novus.dri.oregonstate.edu] ~/Projects/ERC
scratch [SSH: novus.dri.oregonstate.edu] /
Projects [SSH: novus.dri.oregonstate.edu] ~


Welcome
tree_view.py 2
Start
Open...
Clone Git Repository...
Connect to...
Recent
Parasitic_plants [SSH: novus.dri.oregonstate.edu] ~/Projects/ERC
scratch [SSH: novus.dri.oregonstate.edu] /
Projects [SSH: novus.dri.oregonstate.edu] ~


EXTENSIONS:...
pdf
vscode-pdf 6ms
Display pdf file in VS...
tomoki1207
pdf 92K
pdf syntax highlighter
Daniel
Install
Markdown PDF 2M
Convert Markdown to...
yzane
Install
PDF Viewer 909K
Portable document fo...
Mathematic Inc
Install
PDF Preview 189K
Preview Portable Doc...
Analytic Si...
Install
PDF Preview 23K
Welcome
Extension: vscode-pdf
tree_view.py 2
Start
New File...
Open...
Clone Git Repository...
Connect to...
Recent
Parasitic_plants [SSH: novus.dri.oregonstate.edu] ~/Projects/ERC
scratch [SSH: novus.dri.oregonstate.edu] /
Projects [SSH: novus.dri.oregonstate.edu] ~


A
B
C
D
A
B
C
D


Gene of interest
Full genome
1. PCR amplify
Sequencing plasmid
2. Ligate into a plasmid
3. Sanger sequence
the cloned DNA
C A A G G G A G G T G T
240

Genome of interest

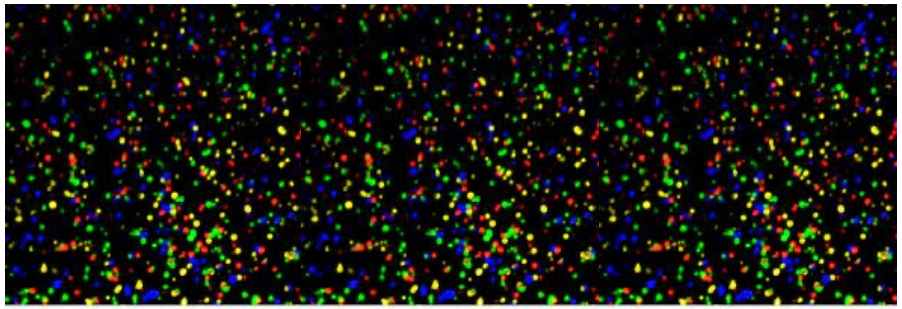
Full genome

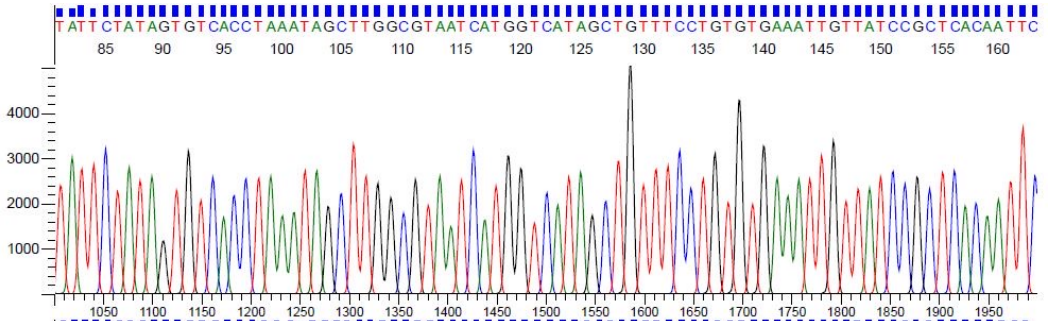
1. Obtain many copies of the genome

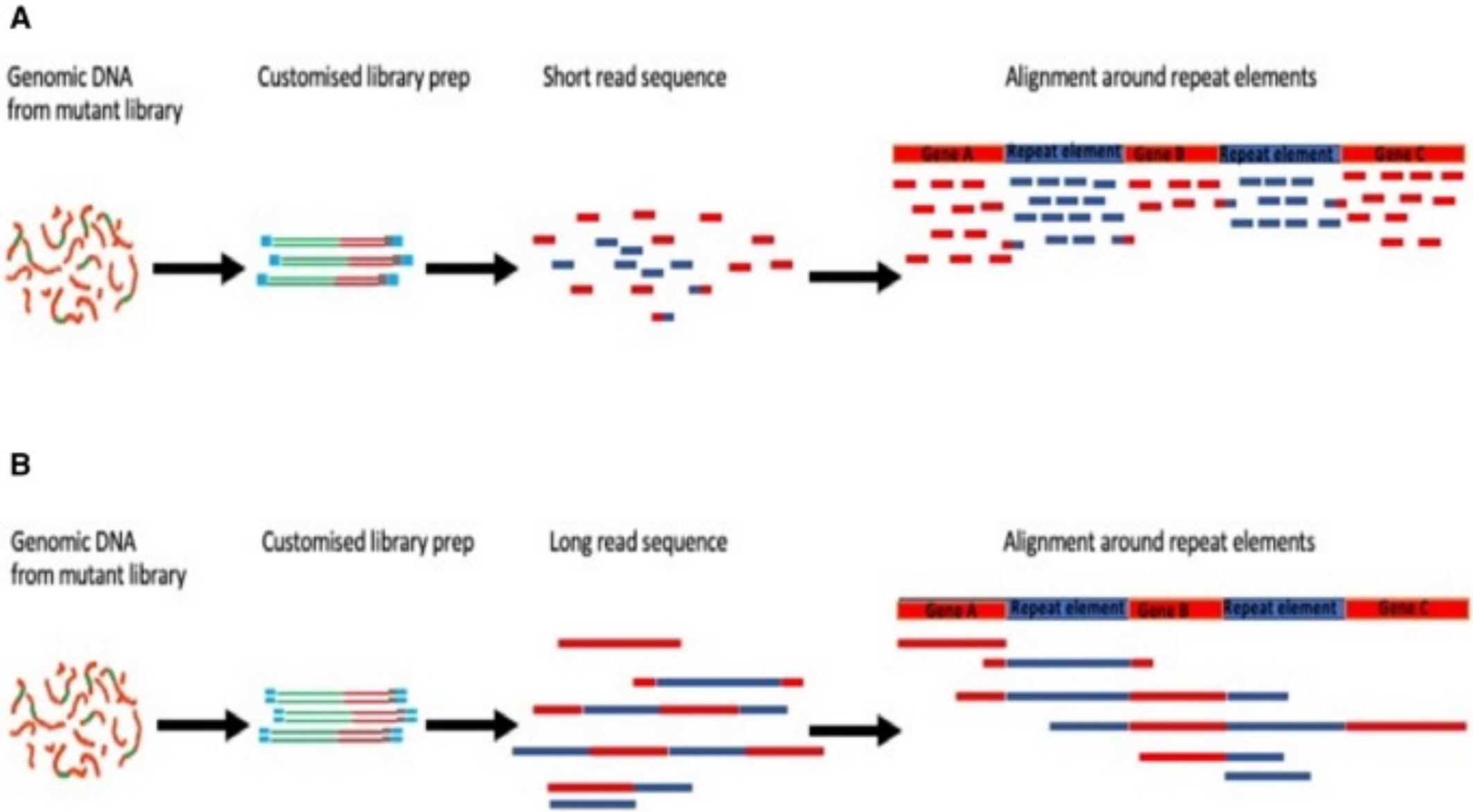
2. Fractionate (break into small pieces)

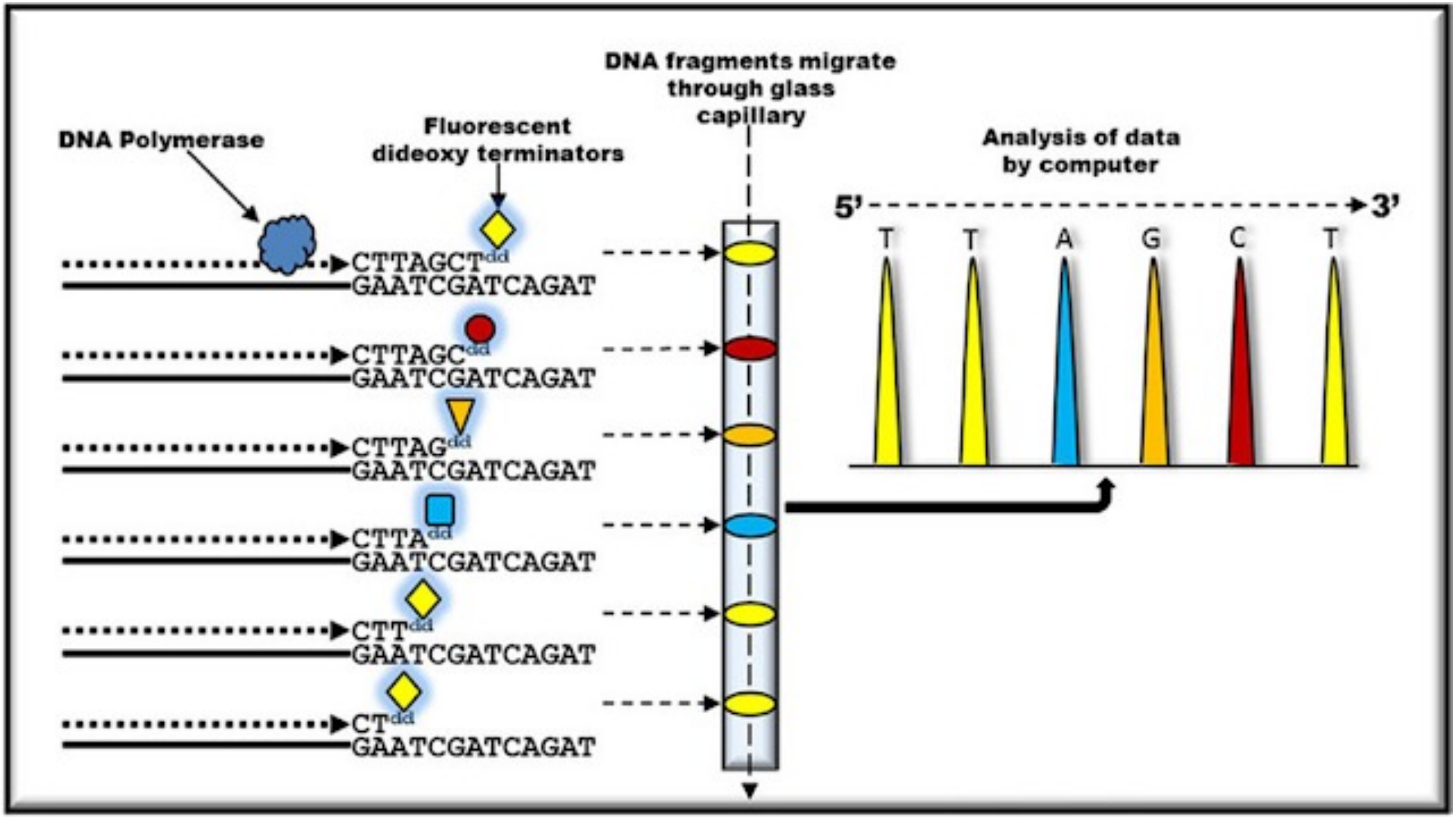
3. Ligate adapters

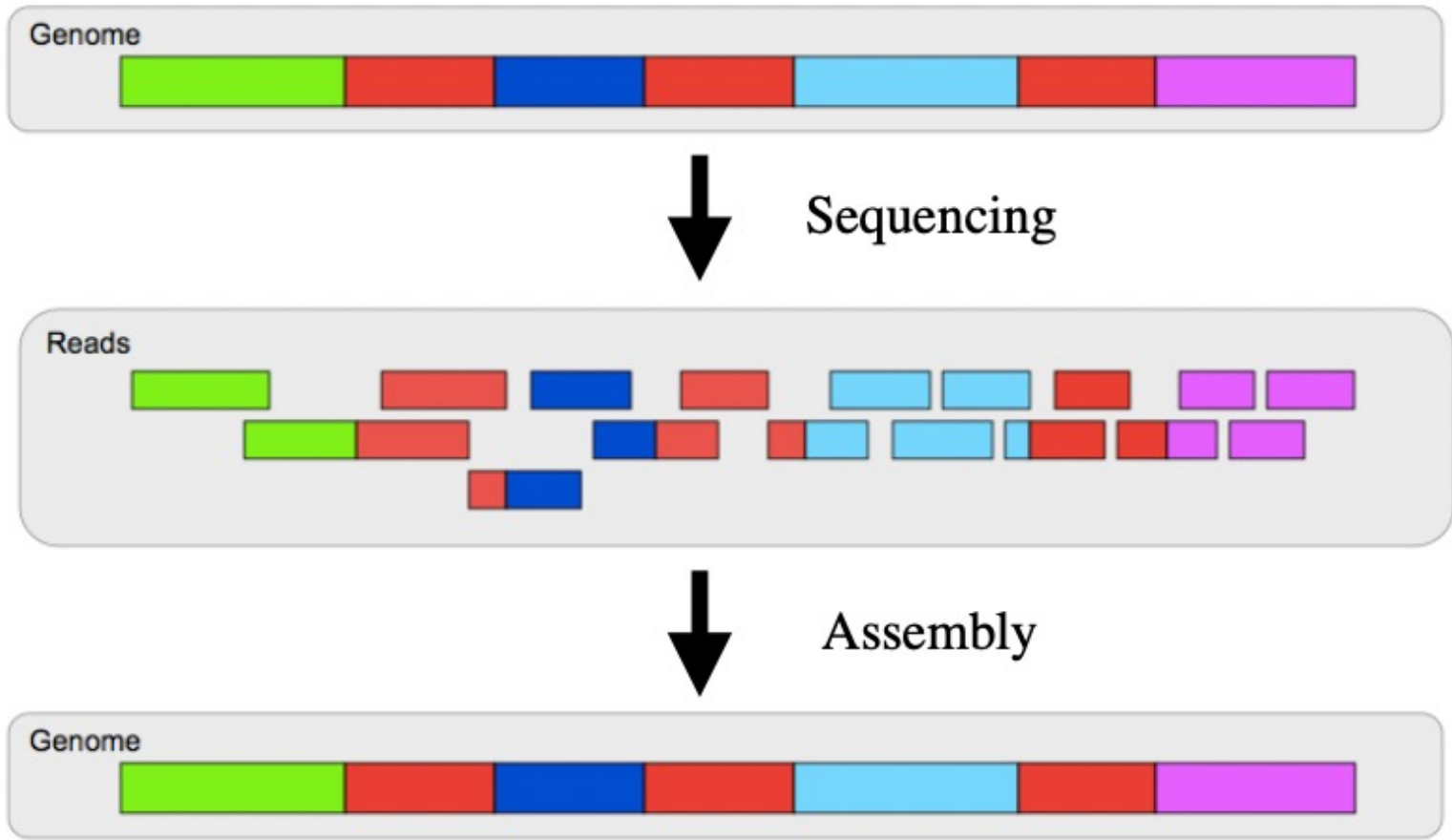
4. Send to next-gen sequencing






A
Genomic DNA from mutant library
Customised library prep
Short read sequence
Alignment around repeat elements
Gene A
Repeat element
Gene B
Repeat element
Gene C
B
Genomic DNA from mutant library
Customised library prep
Long read sequence
Alignment around repeat elements
Gene A
Repeat element
Gene B
Repeat element
Gene C


DNA fragments migrate through glass capillary
DNA Polymerase
Fluorescent dideoxy terminators
Analysis of data by computer
5'
3'
T T A G C T
CTTAGCTddGAATCGATCAGAT
CTTAGCddGAATCGATCAGAT
CTTAGddGAATCGATCAGAT
CTTAddGAATCGATCAGAT
CTTddGAATCGATCAGAT
CTddGAATCGATCAGAT


Genome
Sequencing
Reads
Assembly
Genome

Whole-genome “shotgun” sequencing starts by copying and fragmenting the DNA

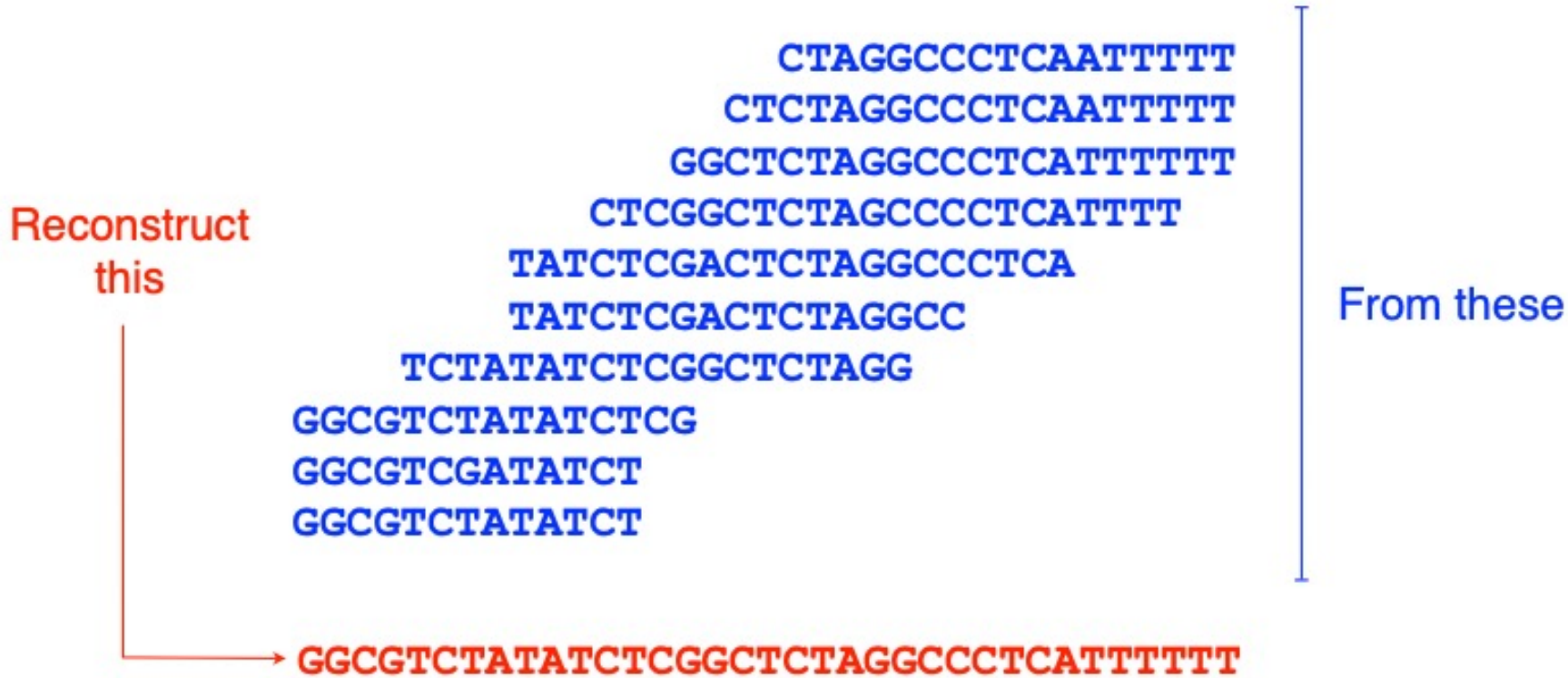
(“Shotgun” refers to the random fragmentation of the whole genome; like it was fired from a shotgun)

Input: GGCGTCTATATCTCGGCTCTAGGCCCTCATTTTTT

Copy:
GGCGTCTATATCTCGGCTCTAGGCCCTCATTTTTT
GGCGTCTATATCTCGGCTCTAGGCCCTCATTTTTT
GGCGTCTATATCTCGGCTCTAGGCCCTCATTTTTT
GGCGTCTATATCTCGGCTCTAGGCCCTCATTTTTT

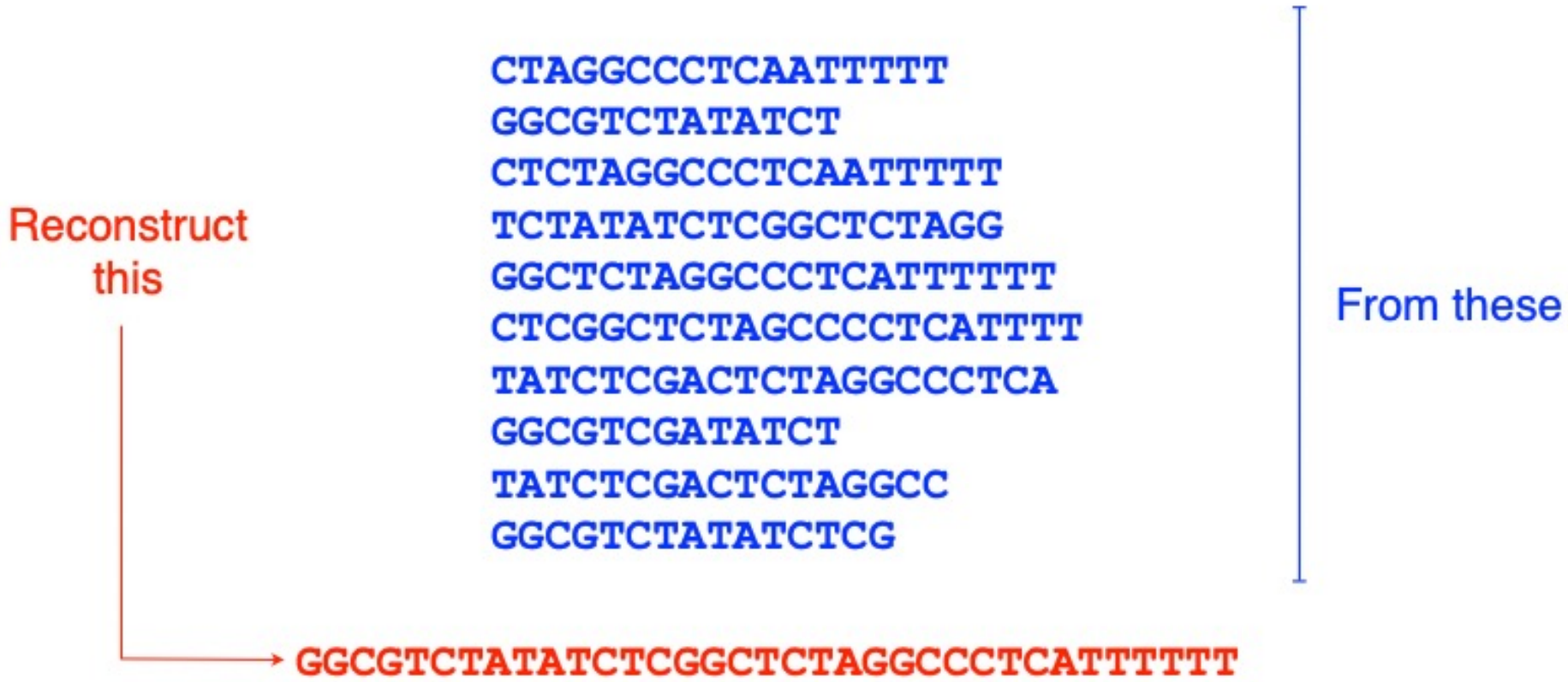
Fragment:
GGCGTCTA TATCTCGG CTCTAGGCCCTC ATTTTTT
GGC GTCTATAT CTCGGCTCTAGGCCCTCA TTTTTT
GGCGTC TATATCT CGGCTCTAGGCCCT CATTTTTT
GGCGTCTAT ATCTCGGCTCTAG GCCCTCA TTTTTT

Assume sequencing produces such a large # fragments that almost all genome positions are covered by many fragments...



JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

...but we don't know what came from where



JOHNS HOPKINS
WHITING SCHOOL
of ENGINEERING

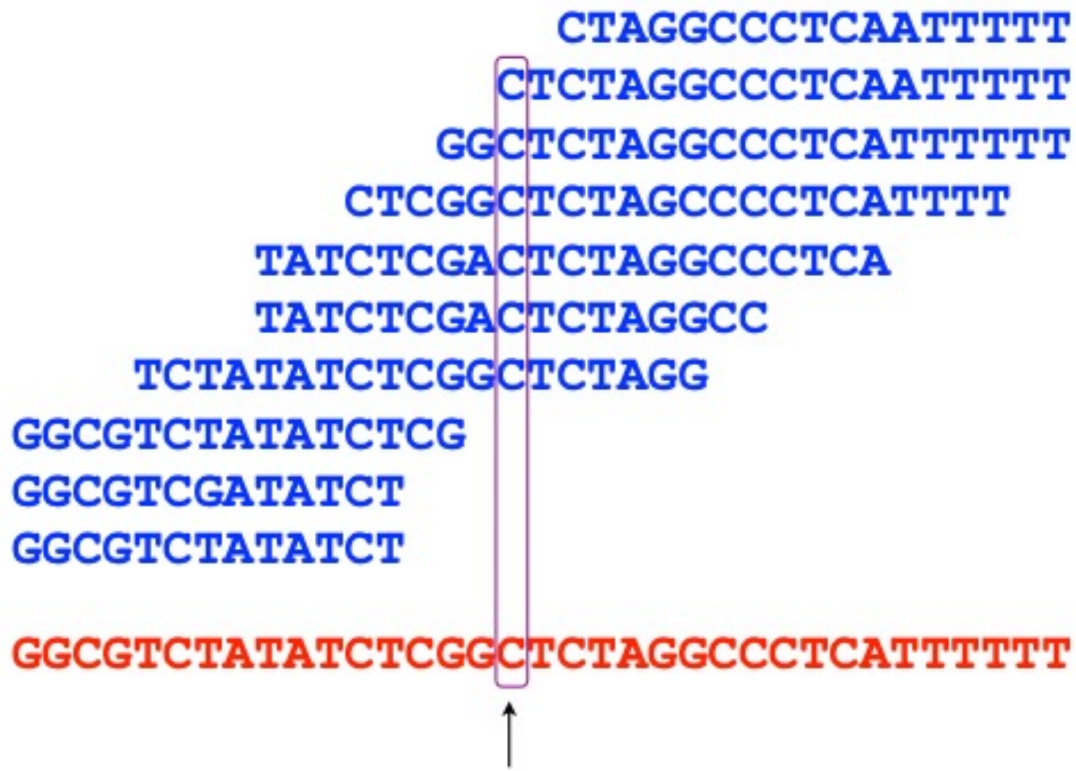
Key term: coverage. Usually it's short for average coverage: the average number of reads covering a position in the genome.

177 nucleotides

35 nucleotides

Average coverage = 177 / 35 ≈ 7x

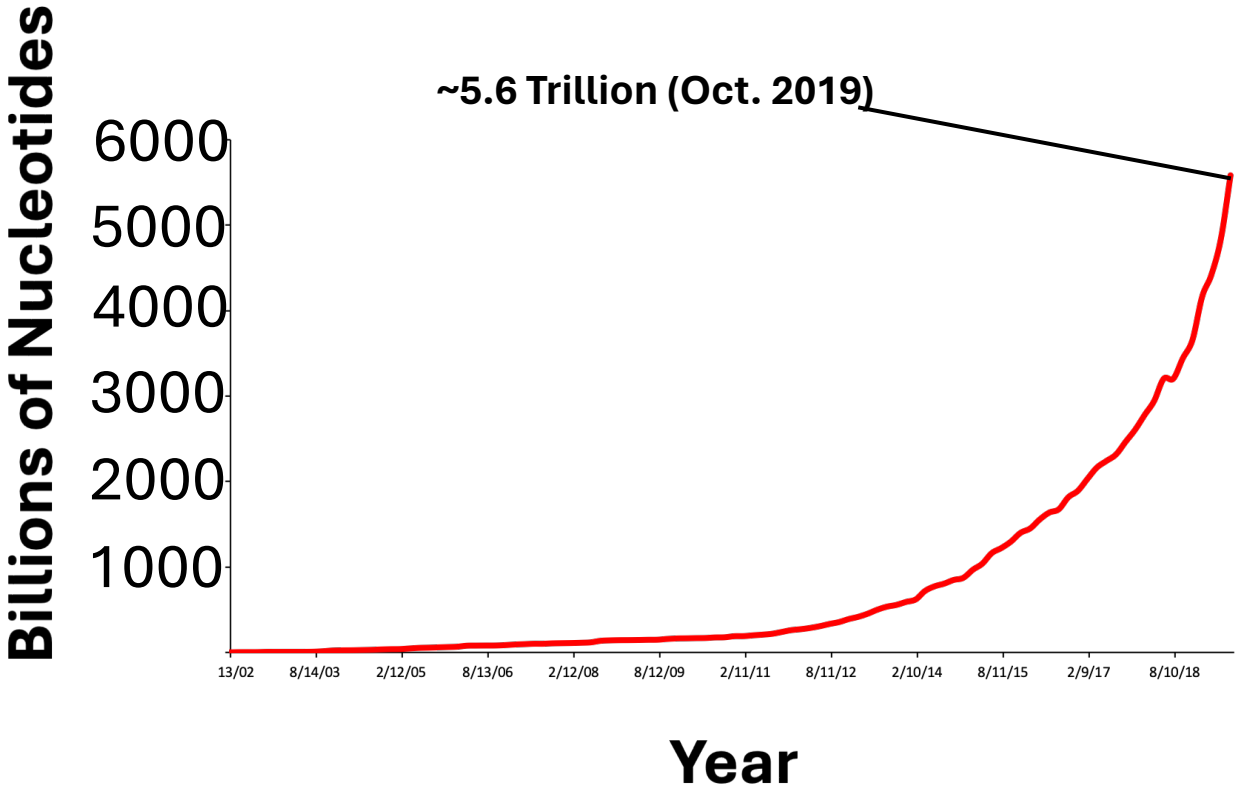
Coverage could also refer to the number of reads covering a particular position in the genome:



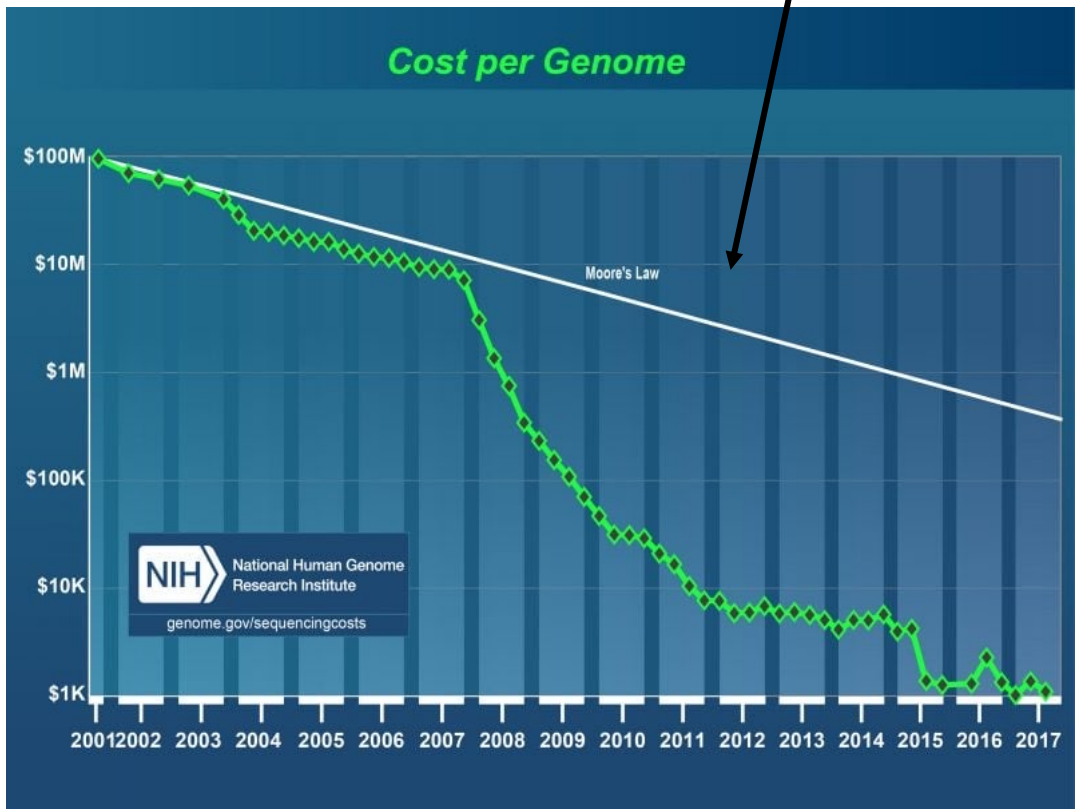
Coverage at this position = 6

Assembly for Short and Long Reads

- Long reads (Pacbio/Nanopore)
  - >10,000bp reads are common
  - Higher error rate (5-15%)
  - **Key computational challenge**: overcome high error rate
- Short reads (Illumina):
  - High accuracy, very high throughput
  - Short read length limits ability to resolve repeats
  - **Key computational challenge**: efficiently assemble large numbers of short reads

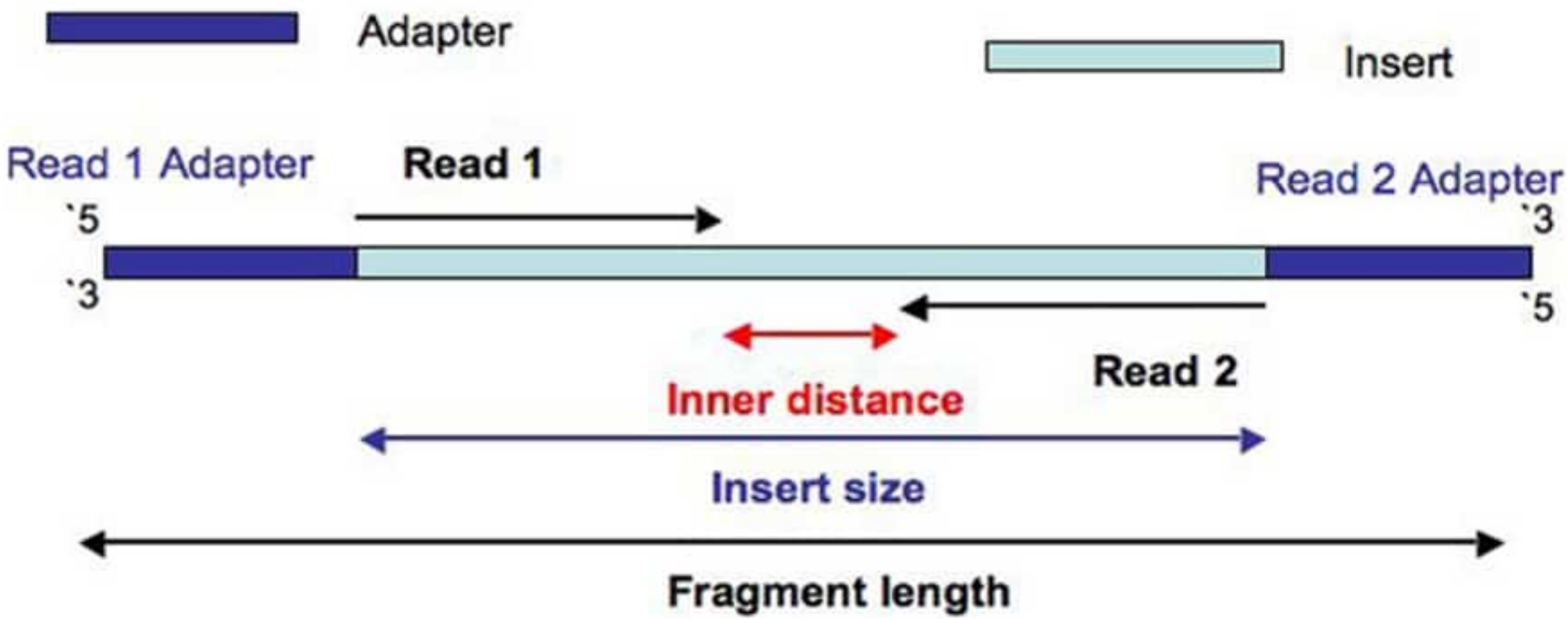


The rate that computing costs have decreased

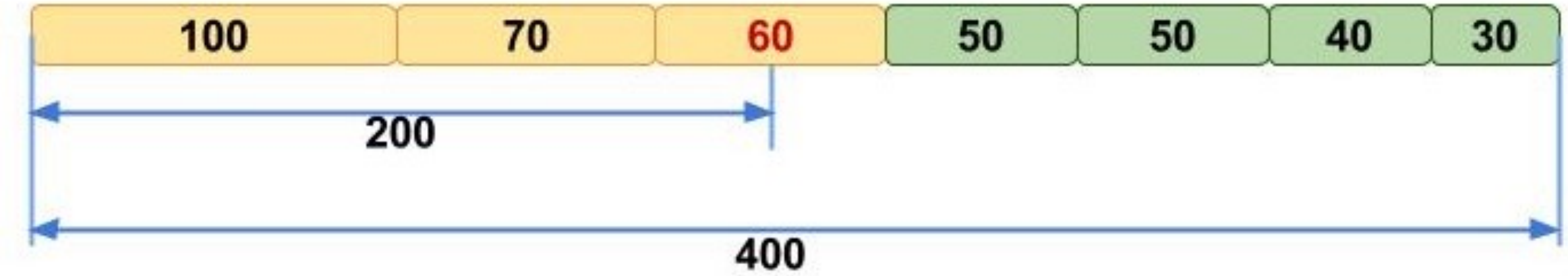


Human genome sequenced:

- Cost more than $2 billion
- 13-year effort by hundreds of scientists


Adapter
Insert
Read 1 Adapter
Read 1
Read 2 Adapter
`5
`3
`3
`5
Read 2
Inner distance
Insert size
Fragment length

N50 (high = good)



Genome fraction of the reference genome (high = good)

Assembly

80% genome fraction

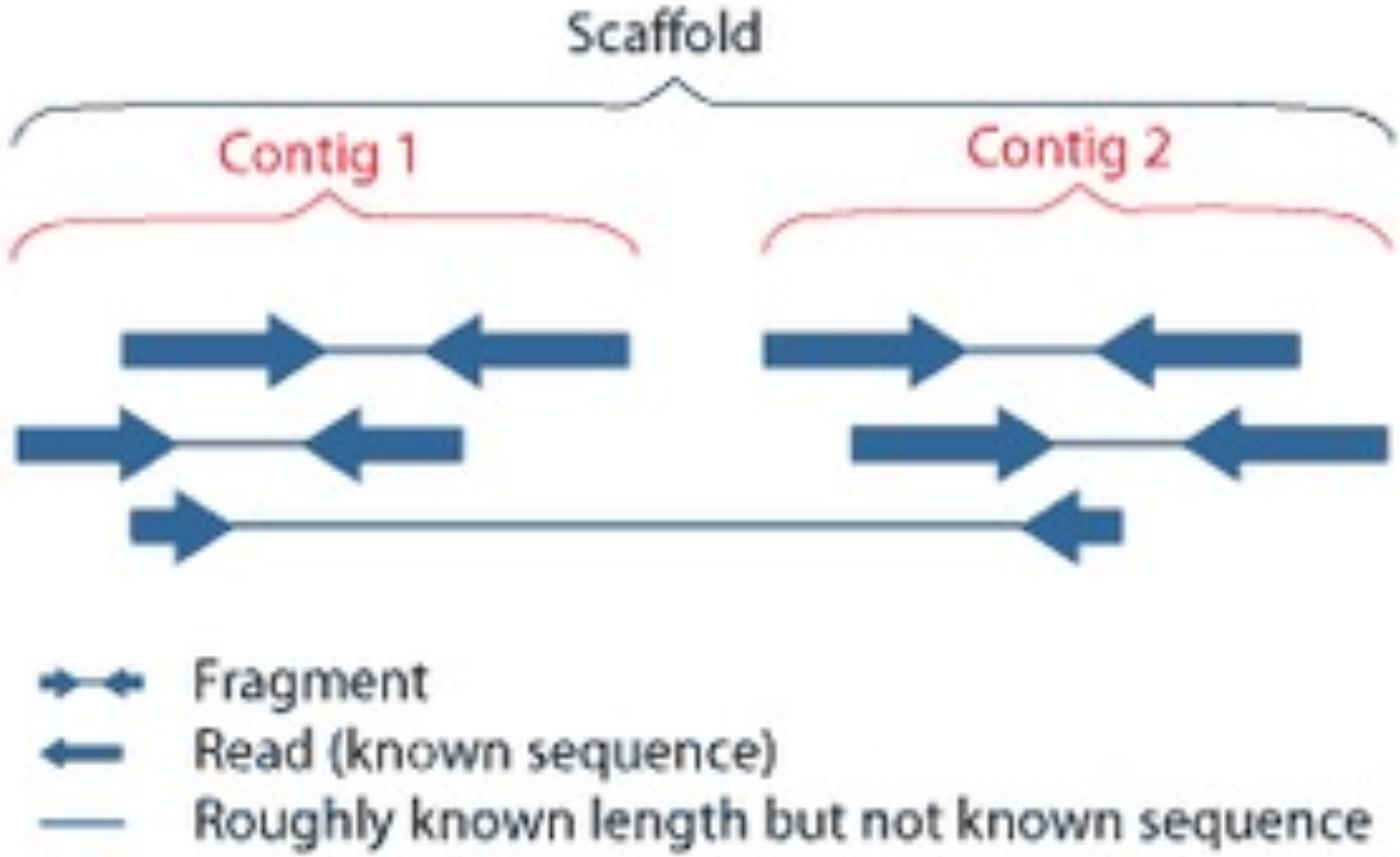
Reference

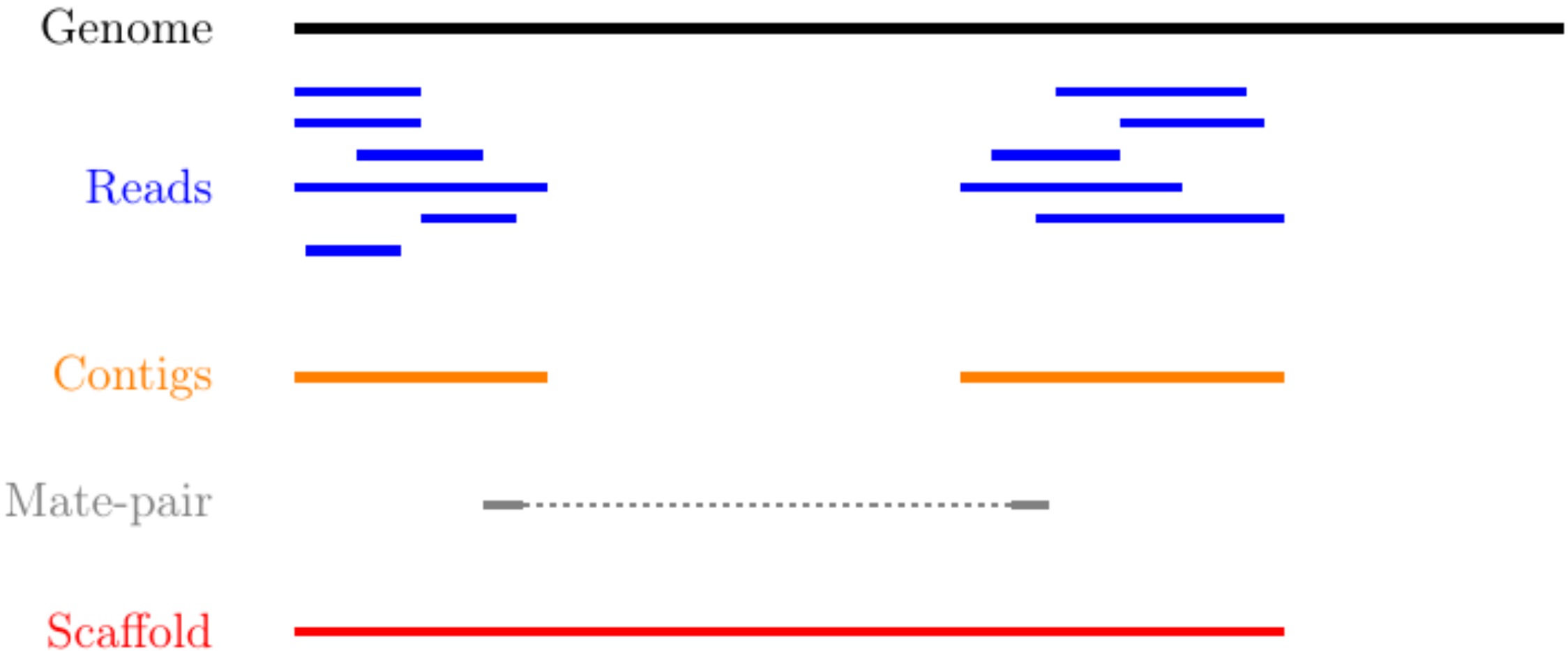
Number of contigs (high = bad)

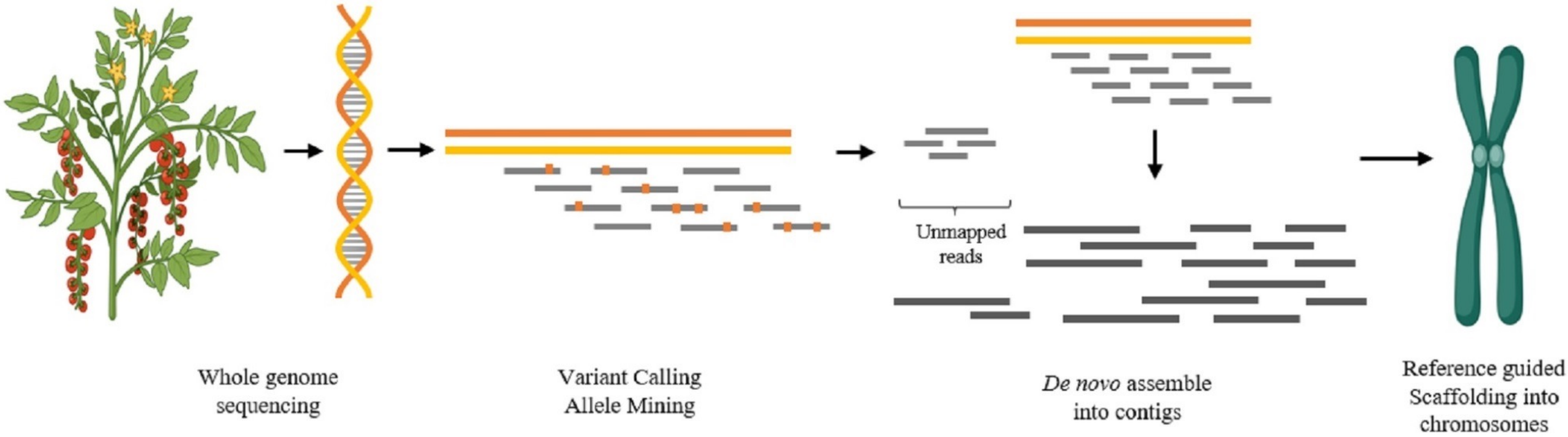
Assembly

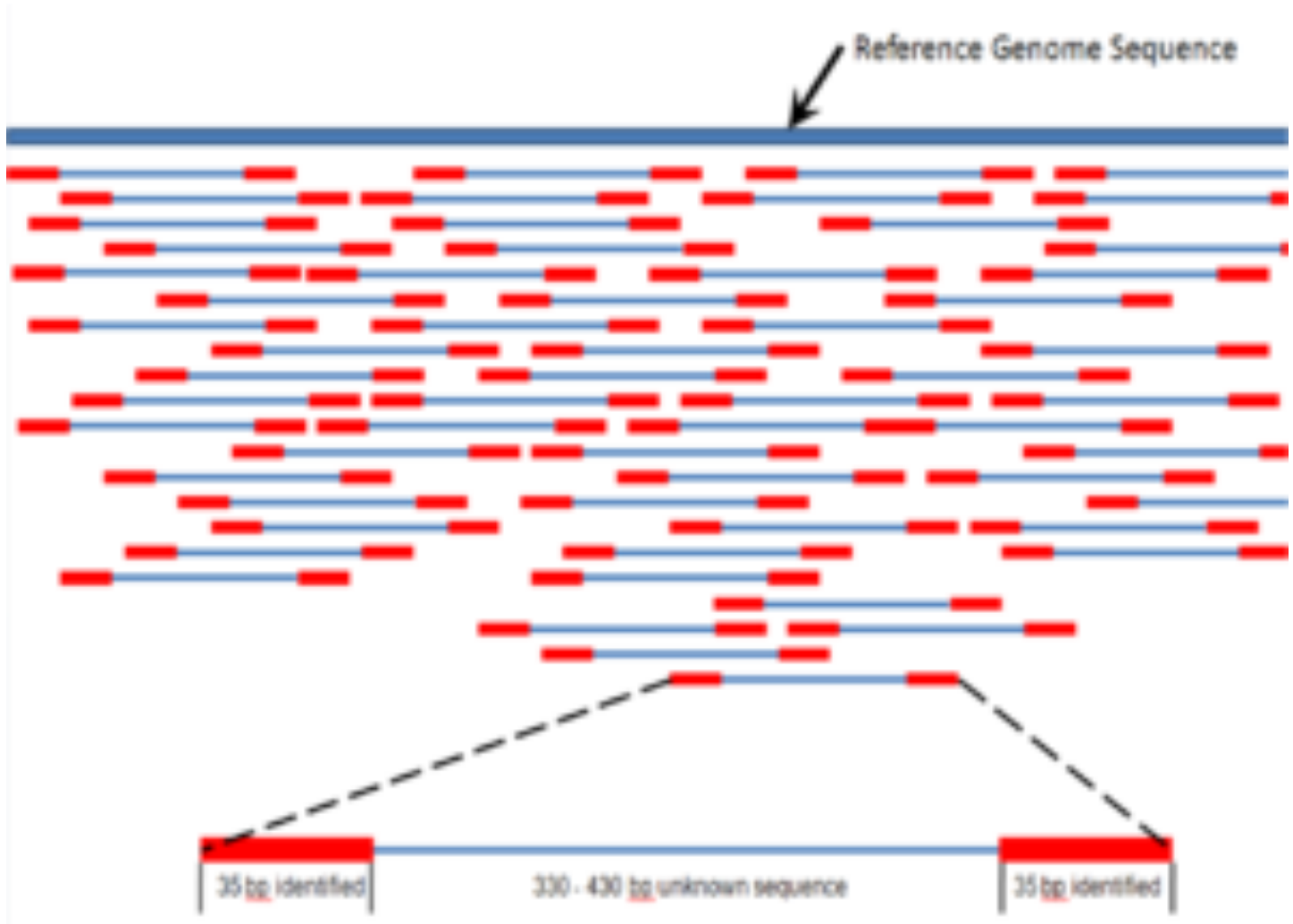
Number of contigs = 5

Reference

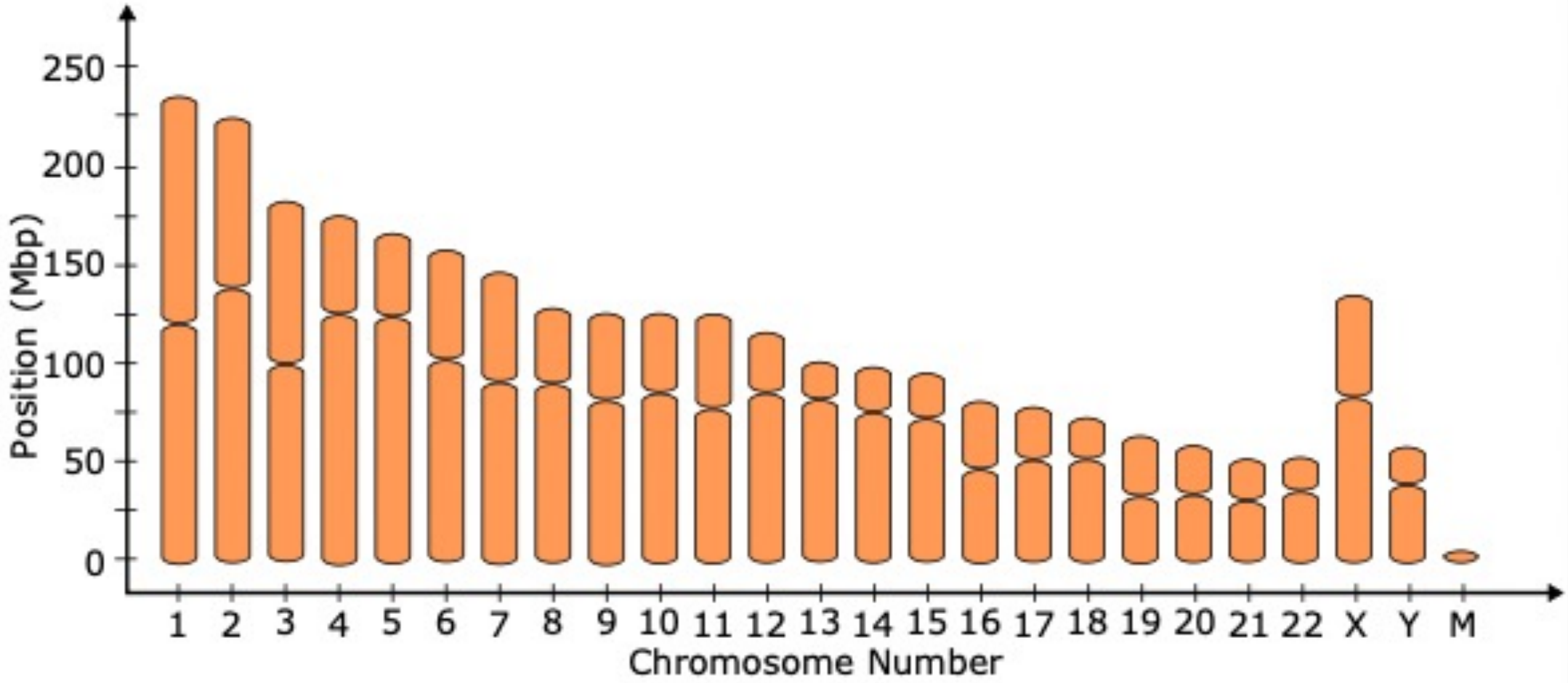

Scaffold
Contig 1
Contig 2
Fragment
Read (known sequence)
Roughly known length but not known sequence


Genome
Reads
Contigs
Mate-pair
Scaffold

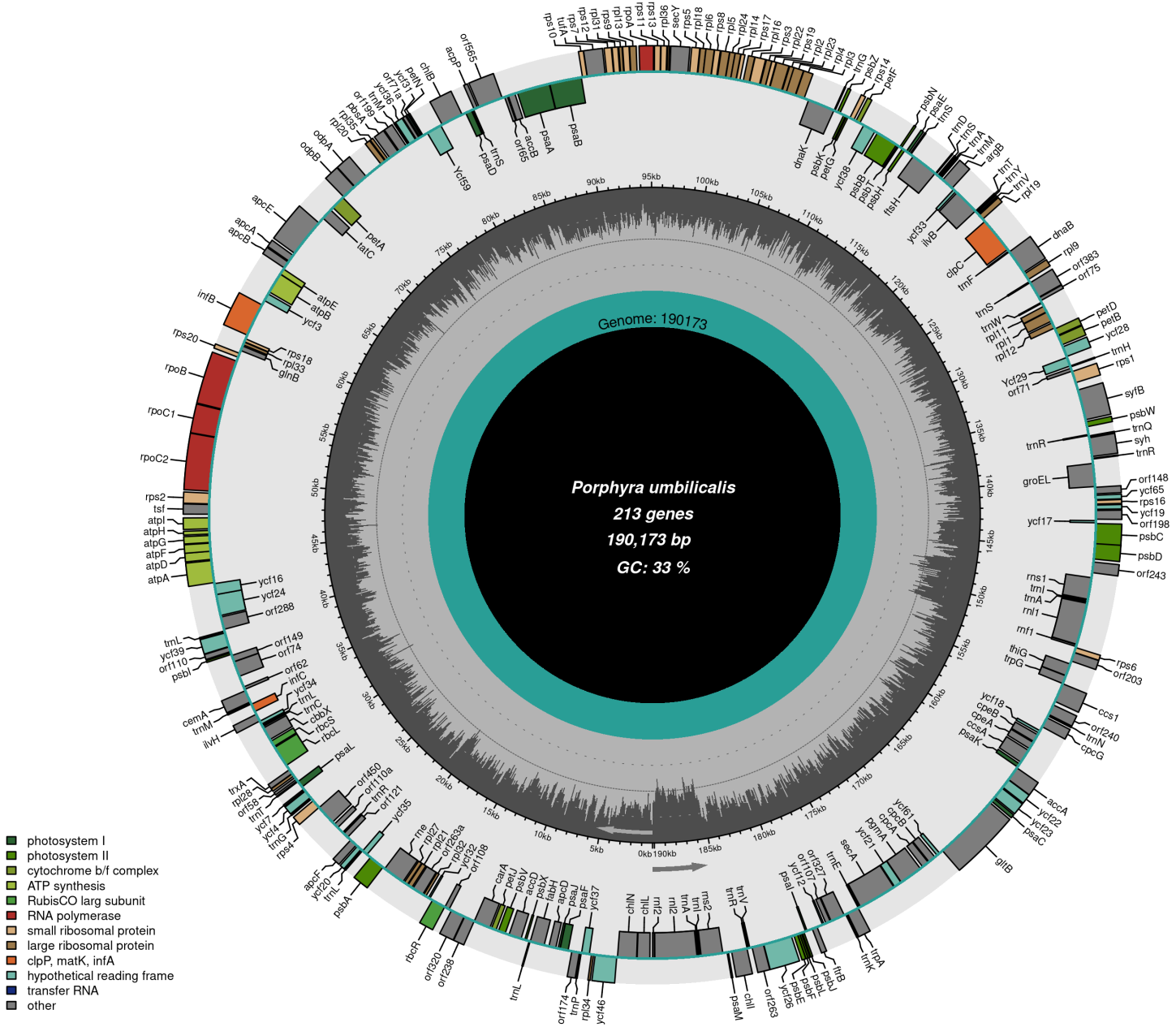

Unmapped
reads
Whole genome
sequencing
Variant Calling
Allele Mining
De novo assemble
into contigs
Reference guided
Scaffolding into
chromosomes


Reference Genome Sequence
35 bp identified
330 - 430 bp unknown sequence
35 bp identified

Week 5


250
200
150
100
50
0
Position (Mbp)
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 X Y M
Chromosome Number


Genome sequence
715,000
720,000
725,000
maker-scf1117875582023-snap-gene-7.50-mRNA-1
maker-scf1117875582023-snap-gene-7.28-mRNA-1
maker-scf1117875582023-snap-gene-7.51-mRNA-1
Gene models
Coding sequences + protein sequences


Genome: 190173
Porphyra umbilicalis
213 genes
190,173 bp
GC: 33 %
photosystem I
photosystem II
cytochrome b/f complex
ATP synthesis
RubisCO larg subunit
RNA polymerase
small ribosomal protein
large ribosomal protein
clpP, matK, infA
hypothetical reading frame
transfer RNA
other

Panda DataFrame objects for working with data tables in python

My_data_tab *in table format*

Columns

Animal	Color
bear	brown
swan	white
tiger	orange
frog	green
crow	black

Rows

Pandas module
for DataFrames in python

My_data_df *in DataFrame format*

Keys

Animal	Color
bear	brown
swan	white
tiger	orange
frog	green
crow	black

Index

```
#Create a dictionary object
animal_dict = {"Animal": ["bear", "swan", "tiger", "frog",
"crow"], "Color": ["brown", "white", "orange", "green",
"black"]}
```

```
#Import the pandas module and store as pd for short
import pandas as pd
```

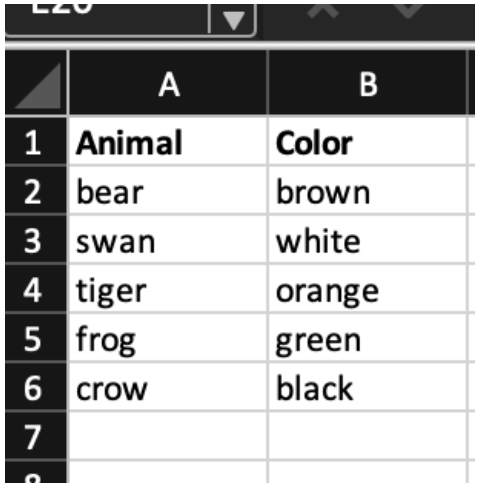
```
#Convert from dictionary to a dataframe object
#using the pandas function .DataFrame()
animal_df= pd.DataFrame(aminal_dict)

print(animal_df)

 Animal   Color
0   bear   brown
1   swan   white
2  tiger  orange
3   frog   green
4   crow   black
```

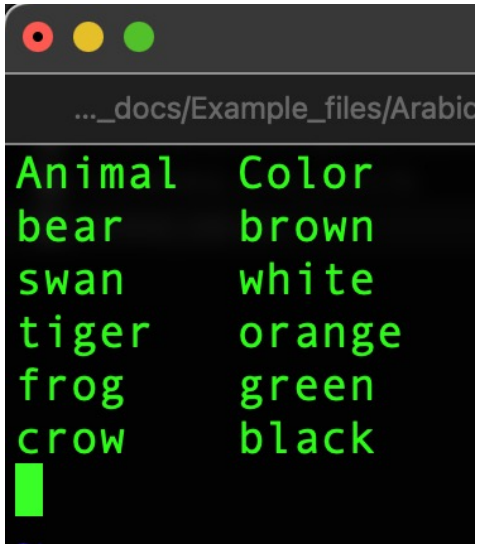
Tab-separated values (.tsv) format stores tables in plain-text

Looking at a table in Excel



	A	B
1	**Animal**	**Color**
2	bear	brown
3	swan	white
4	tiger	orange
5	frog	green
6	crow	black
7		

Looking at a table in Vim



Looking at the plain text for a table

```
Animal  Color
bear    brown
swan    white
tiger   orange
frog    green
crow    black
```

There are 'tab symbols' between the items on each line

- Tab symbols are encoded by "\t" in python (and many other languages)
- Tab symbols tell programs like Excel and Pandas which values belong in separate fields
  - **Tab Separated Values (.tsv) format** is an extremely common format for storing table data

```
#Specify the Full path to Animal_data.tsv
animal_file_path="/home/hub_data_share/Examples/Mod06/Animal_tab.tsv"
```

```
# Read in the Animal_data.tsv file
my_df=pd.read_csv(animal_file_path, delimiter= "\t")
```

```
## Explore the dataframe a little bit

#Get a list of all the column headers
print(list(my_df.keys()))

#Get the number of rows in the df
print(len(my_df.index))
```

GFF3 format files store gene annotation data in tsv format

```
>Chr5
TATACCATGTACCCTCAACCTTAAAACCCTAAAACCTATACTATAAATCTTTAAAACCTATACTCTAAACCATAGGGTT
TGTGAGTTTGCATAAAGTGTCACGTATAAGTGTTTCTAACATGTGAGTTTGCATAAGAGTCTCGACTATGTGTTTGTTC
AAAAGTGACGTAAGTGTTTAGACTAGAGCCGGCCGTGAGCACAAGCGGGCCAAGCCCATGCTTGCGGCAGATTATCCTA
TAATTATGTTTTGCGGCTTTACAATTTTGAATTTGTTTTGTTGTTTAGTGGTCGAGTCAGGGATGAGTTTGTTCTCAAA
CATCTCAAATTTCTAATCTTCCAAGTGTTAGGTTCACTCTACCTTATTTTTTTTTTTTTTTTTATTTTAAATTTCTAT
TAGAAAGAACTTGAACCTTATTTCAAGTGAACATTAAAAAGAATTGTAACCCTACCTTATTTCTATTATTATGAATATG
TGTAAAGTAAATTTTTTTACAATAAGCATGTATAAAGTAACATATTTTTGTTCTTTTCTTATAGCCTTGTAAATCATAG
GGACGGATCTAGTTTAGAATAAAGTGTTGCATAAGTTCTTCGACTAACATGTGAGTCCATTTTTGTGAGTTTTGCATAA
CTTTTGCGCCTTCGACAATTTGAATTTTATGCATAAGTGCTTTGACTAATTTATATGTTTTGTCTAAGTGATTCGACAT
TTCGACTAAGTGATGTGAATTGTGTCATTCGTCGTCGTTGTCGCATCCGGAAGCGTGTTTGTGGTTCTTGTGTCTTACT
TGCCGGGTTATGAAGTCCTTTGTTTCGAGAGATCTGCTCTCATCATGATAGAAGATGGTAGATGTTATCTTTGGATTGC
CGCGGGTTGAATTGTCGCGGGCATCCTTCTTAAATCTCATTCACATGTTCTTTTGAGGAAGAGTTTTTTTTGTTTTTTA
TTTATTTATTGTTTCCCTTAAGATTTATACTTTATTCCTTCTATTTTTTGAGGACTTTTATTTTGTCTGTTGGCTTTAG
```

A genome sequence in fasta format:

- How to do we know where the genes are??

GFF3 (.tsv) format

seqid	source	type	start	end	strand	attributes
Chr5	phytozomev12	gene	995	5156	-	ID=AT5G01010.Araport11.447;Name=AT5G01010
Chr5	phytozomev12	mRNA	995	4994	-	ID=AT5G01010.2.Araport11.447;Name=AT5G01010.2;pacid=37423170;longest=1;Parent=AT5G01010.Araport11.447
Chr5	phytozomev12	CDS	4765	4924	-	ID=AT5G01010.2.Araport11.447.CDS.1;Parent=AT5G01010.2.Araport11.447;pacid=37423170
Chr5	phytozomev12	five_prime_UTR	4925	4994	-	ID=AT5G01010.2.Araport11.447.five_prime_UTR.1;Parent=AT5G01010.2.Araport11.447;pacid=37423170
Chr5	phytozomev12	CDS	4552	4679	-	ID=AT5G01010.2.Araport11.447.CDS.2;Parent=AT5G01010.2.Araport11.447;pacid=37423170
Chr5	phytozomev12	CDS	4335	4467	-	ID=AT5G01010.2.Araport11.447.CDS.3;Parent=AT5G01010.2.Araport11.447;pacid=37423170
Chr5	phytozomev12	CDS	4102	4258	-	ID=AT5G01010.2.Araport11.447.CDS.4;Parent=AT5G01010.2.Araport11.447;pacid=37423170
Chr5	phytozomev12	CDS	3927	4005	-	ID=AT5G01010.2.Araport11.447.CDS.5;Parent=AT5G01010.2.Araport11.447;pacid=37423170
Chr5	phytozomev12	CDS	3762	3802	-	ID=AT5G01010.2.Araport11.447.CDS.6;Parent=AT5G01010.2.Araport11.447;pacid=37423170
Chr5	phytozomev12	CDS	3543	3659	-	ID=AT5G01010.2.Araport11.447.CDS.7;Parent=AT5G01010.2.Araport11.447;pacid=37423170