

Finding Orthologs and Paralogs

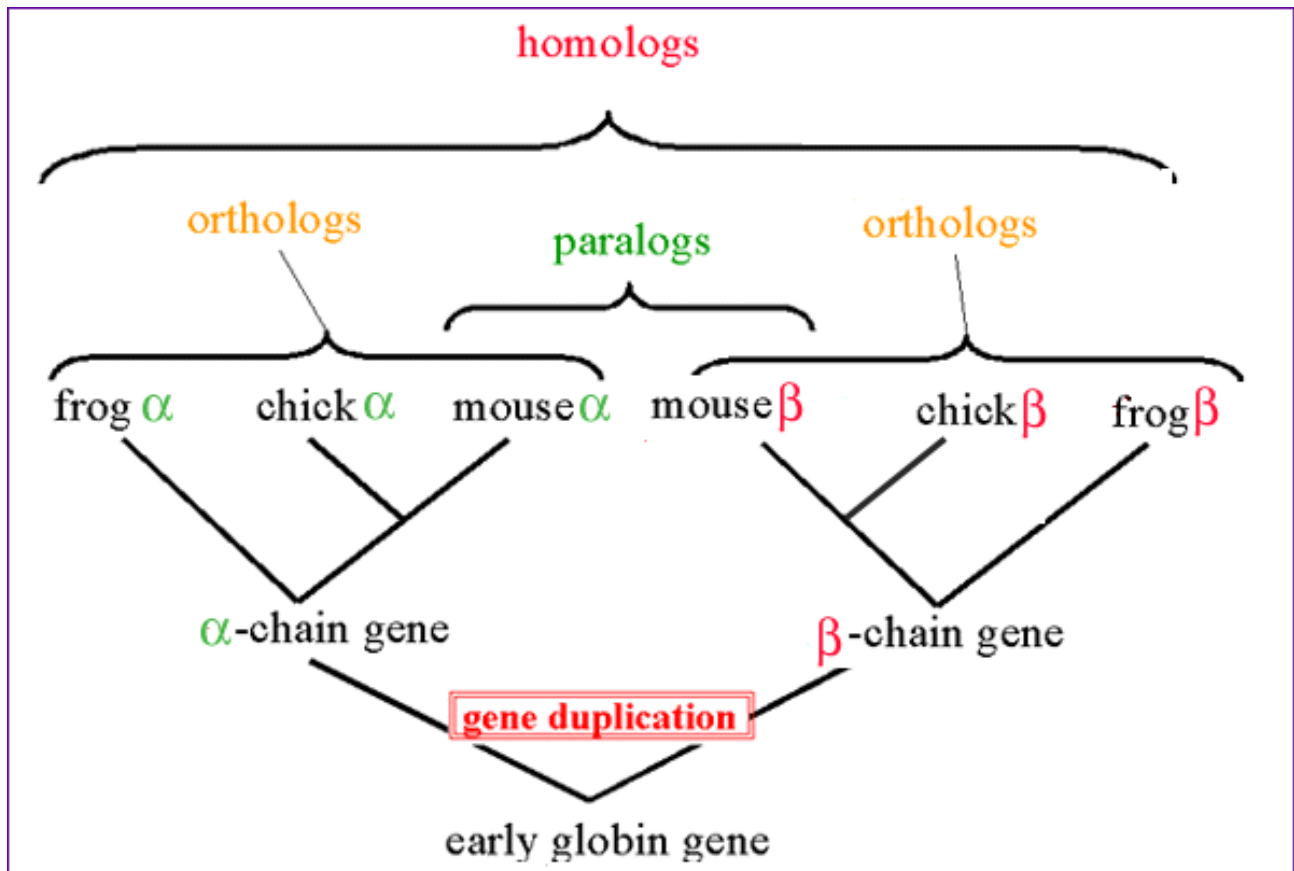


Figure 1: Orthologs

Gene families and Orthology

Problem: How to find “same” genes across multiple species.

Genes can duplicate (Paralogs) and can be identical due to descent (Ortholog)

Methods

- BLAST: 1 way BLAST (Gene A in Species X, what is best hit in Species Y)
- BLAST: reciprocal BLAST

Trees can help resolve relationships

Best hits can sometimes be wrong (B) though it can be resolved with phylogenetics.

Reciprocal Searches

- Bi-directional or Reciprocal BLAST

Implement Bidirectional

Method to find best top hit in one direction and the reverse.

Let's walk through the [code](#)

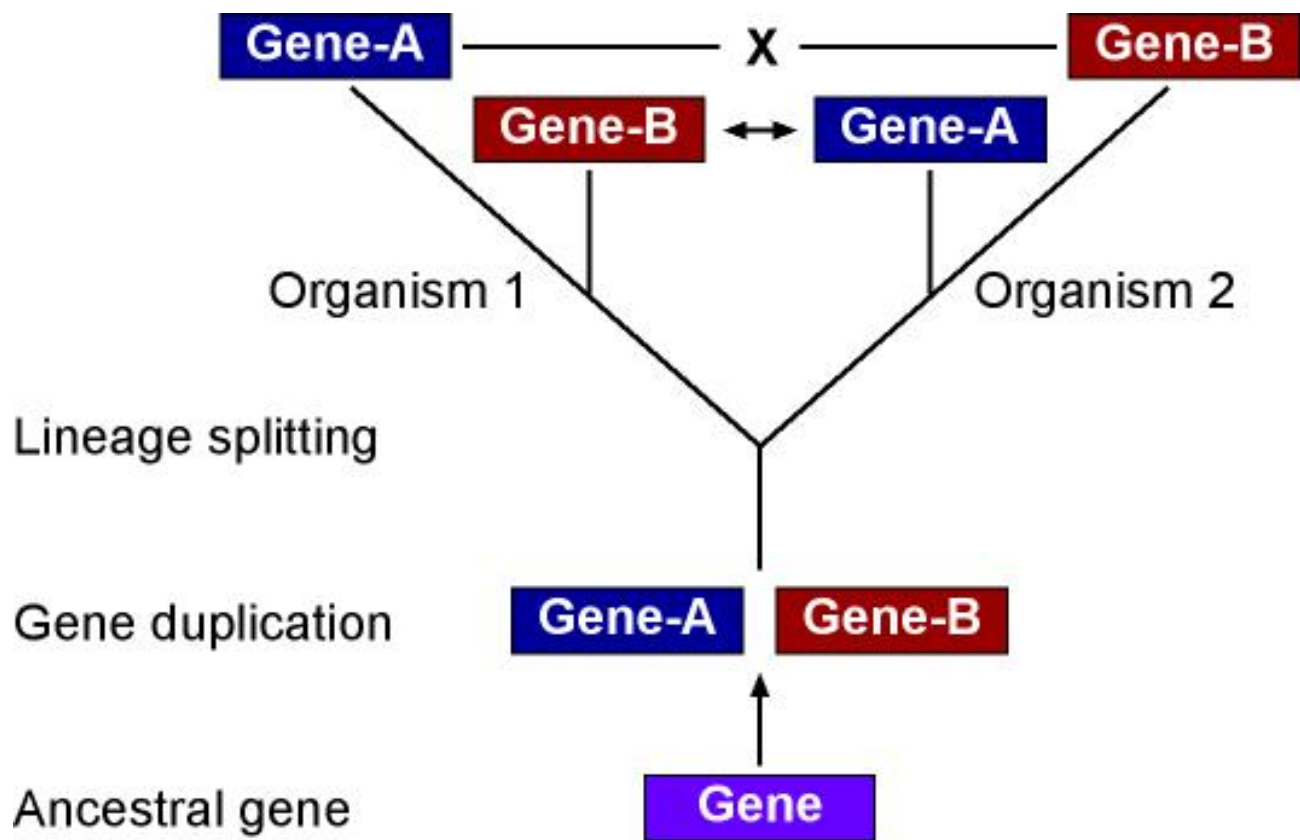


Figure 2: orthologs

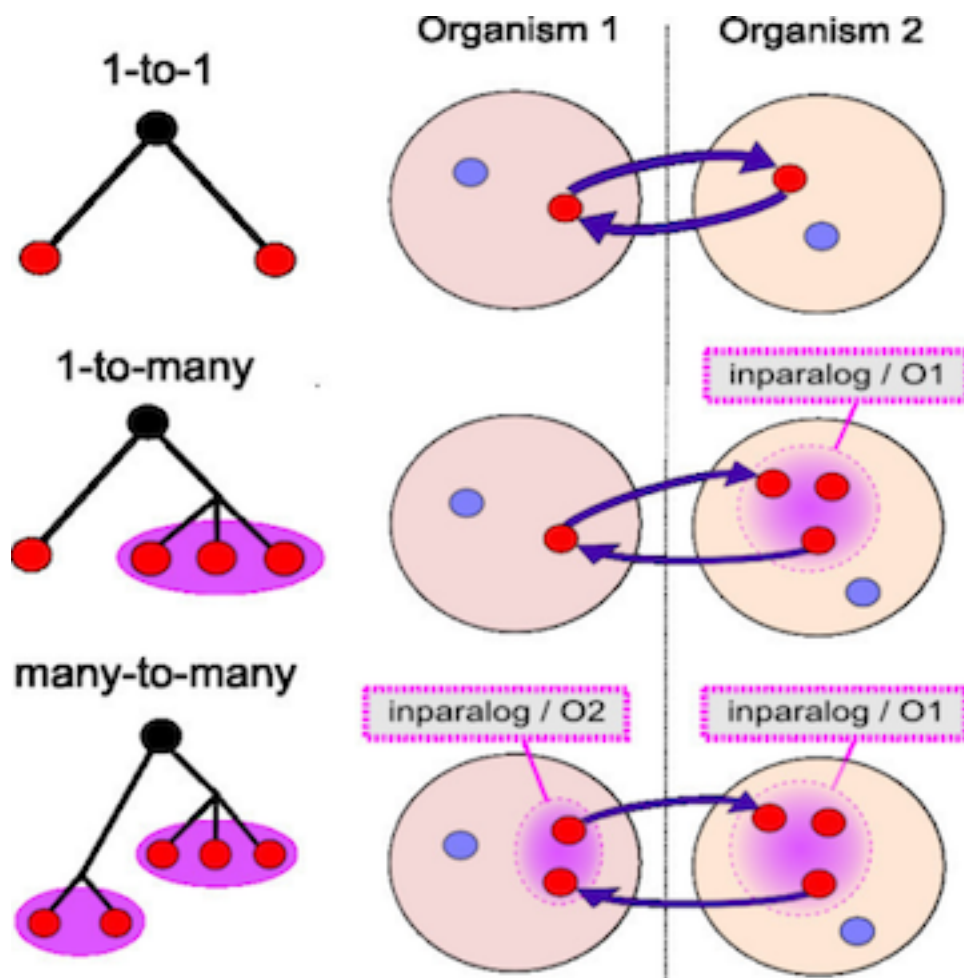


Figure 3: diagramorth

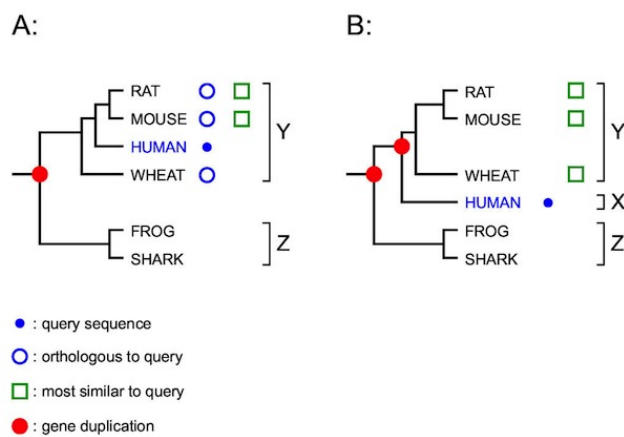


Figure 4: RIO

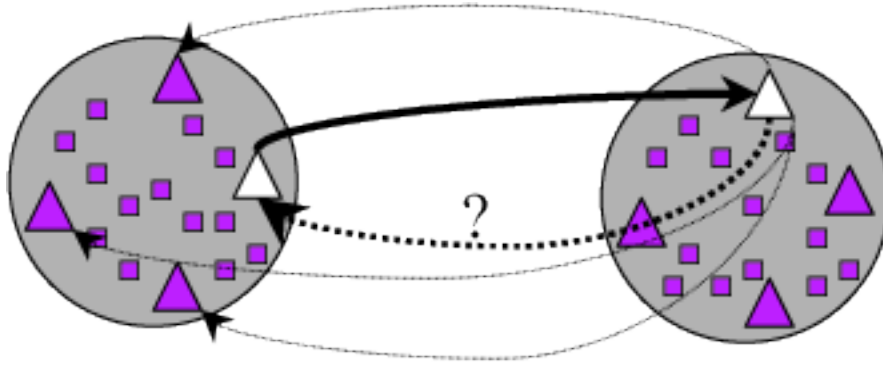


Figure 5: BRH

Will write this in Python in Class

Clustering

- Lumping genes together based on similarity linkage
- Single-linkage means if there is a link between A-B then they are in a cluster

Code up single-linkage

Let's look at some [code](#).

Will write this in Python in Class

Issues

Tools to go after Orthologous and Paralogous Sequences

- [OrthoFinder](#)

Steps to build orthologs on cluster

We will take 3 datasets of annotated Cyanobacteria, download and run analysis to generate Ortholog table.

```
#!/bin/bash -l
#SBATCH -p short -N 1 -n 1 -c 8 --mem 8G --time 2:00:00
#SBATCH -J orthofinder
#SBATCH -o logs/%x.%j.log
module load ncbi-blast
module load orthofinder

set -euo pipefail
CPU=${SLURM_CPUS_PER_TASK:-1}

mkdir -p cyanobacteria
cd cyanobacteria
curl -L -O ftp://ftp.ensemblgenomes.org/pub/bacteria/release-45/fasta/bacteria_10_collection/oscillatoria
curl -L -O ftp://ftp.ensemblgenomes.org/pub/bacteria/release-45/fasta/bacteria_0_collection/nostoc_puncti
curl -L -O ftp://ftp.ensemblgenomes.org/pub/bacteria/release-45/fasta/bacteria_4_collection/cyanobacteriu

# uncompress files and name them all *.fasta
for file in *.fa.gz
```

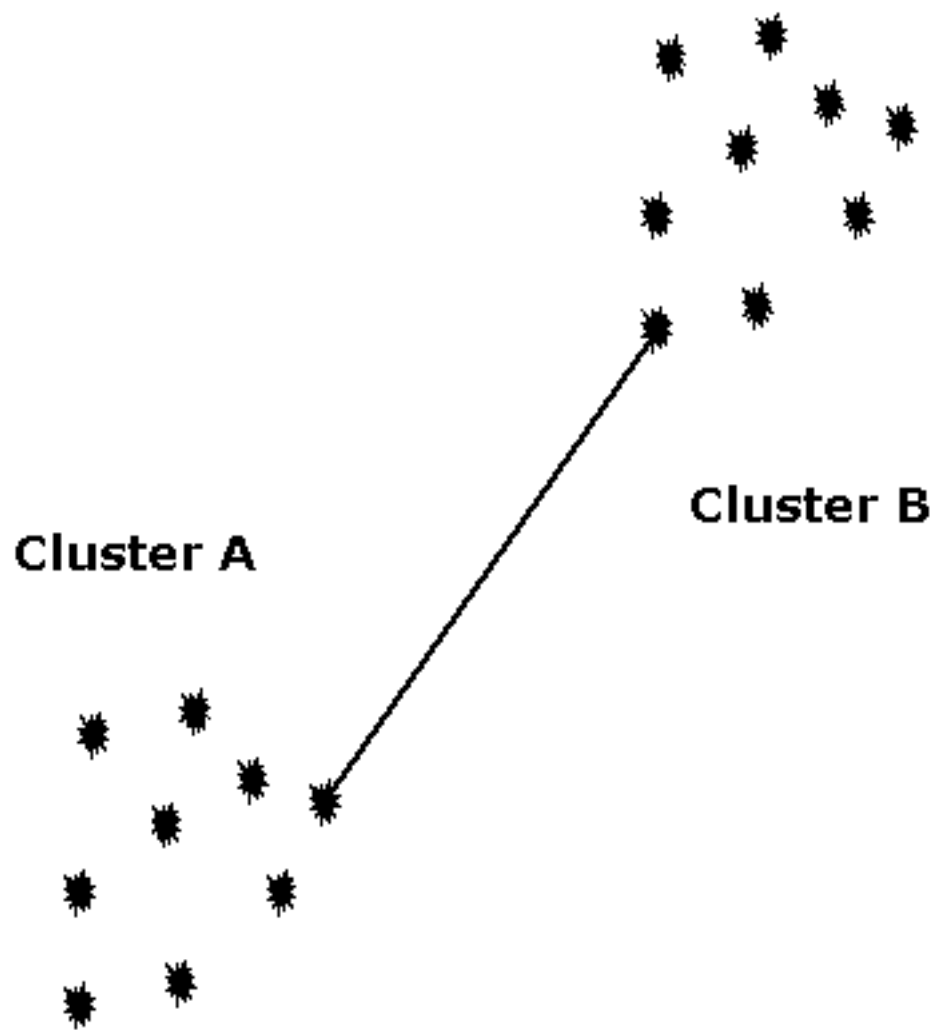


Figure 6: SingleLinkage

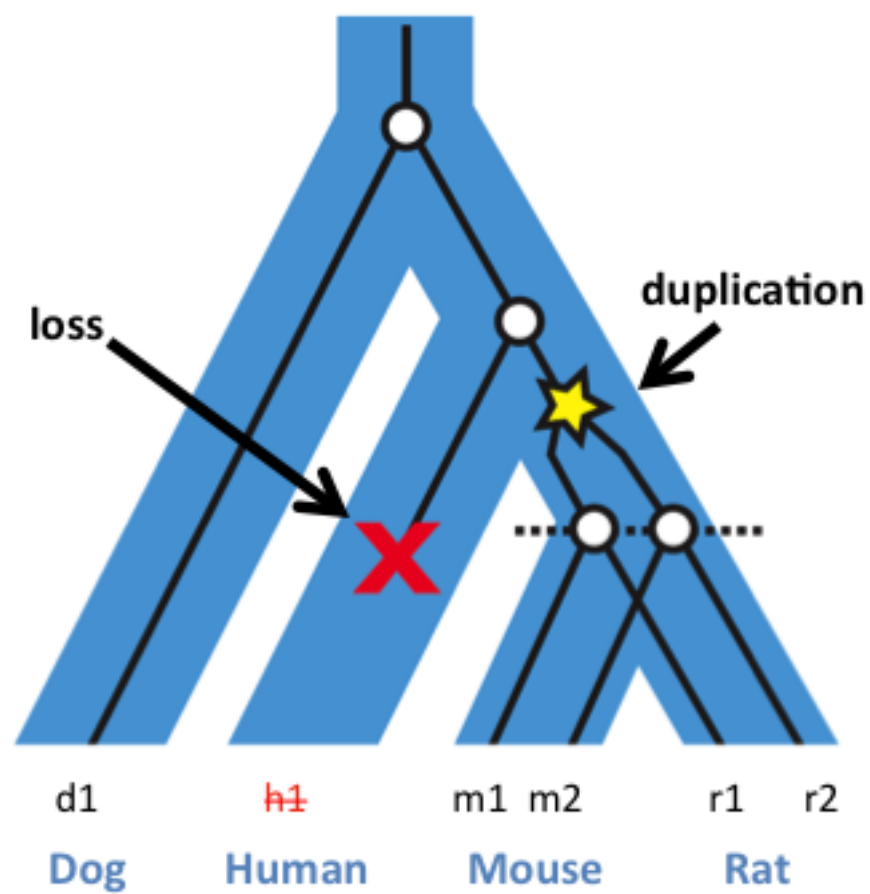


Figure 7: orthologsloss

```
do
  m=$(basename $file .pep.all.fa.gz)
  pigz -dc $file > $m.fasta
done

cd ..

# -t: threads for the all-vs-all sequence searches, -a: threads for the tree/analysis steps
orthofinder -t $CPU -a $CPU -f cyanobacteria
```

Run `mkdir -p logs` before you `sbatch` it. See [UNIX IV](#) for how to choose `-c`, `--mem` and `--time`.

Ortholog results

The output file by default will be the date of the analysis. Opening the file `cyanobacteria/Results_XXX/Orthogroups.txt` but I made a [folder](#) in the examples you can look over. Here's one [table](#)

Format

GroupName\tSp1_Gene1, Sp1_Gene2\tSp2_Gene1, Sp2_Gene2\tSp3_Gene1, Sp3_Gene2

```
Cyanobacterium_aponinum_pcc_10605.ASM31767v1    Nostoc_punctiforme_pcc_73102.ASM2002v1    Oscillatoria
OG00000000      EKQ66605, EKQ66611, EKQ66662, EKQ66782, EKQ66954, EKQ66984, EKQ67084, EKQ67085, EKQ67
EKQ67433, EKQ67590, EKQ67680, EKQ67799, EKQ67807, EKQ67983, EKQ68026, EKQ68032, EKQ68054, EKQ68173, EKQ6
EKQ69279, EKQ69300, EKQ69345, EKQ69368, EKQ69506, EKQ69549, EKQ69629, EKQ69630, EKQ69655, EKQ69739, EKQ69
KQ70786, EKQ70840, EKQ70870, EKQ70894, EKQ71088, EKQ71090, EKQ71265, EKQ71335
OG00000001      AFZ52442, AFZ54265, AFZ54640      ACC78968, ACC78978, ACC79054, ACC79090, ACC79138, ACC79182, A
81797, ACC82091, ACC82628, ACC82978, ACC83035, ACC83215, ACC83711, ACC84528, ACC84844, ACC84851, ACC85118
69971, EKQ69995, EKQ70003, EKQ70556, EKQ70833, EKQ71286
OG00000002      AFZ55137      ACC79344, ACC80485, ACC80595, ACC82143, ACC82836, ACC82962, ACC83845, ACC83999, A
ACC84972, ACC84974, ACC84981, ACC84982, ACC84983, ACC85032 EKQ66950, EKQ67597, EKQ67615, EKQ68095, EKQ68
OG00000003      AFZ53198      ACC78875, ACC78976, ACC79256, ACC79524, ACC79759, ACC80145, ACC80528, ACC80591, A
ACC82769, ACC83025, ACC83081, ACC83457, ACC83602, ACC83721, ACC83749, ACC84422, ACC85331      EKQ69908
OG00000004      ACC80422, ACC80525, ACC80662, ACC80851, ACC80857, ACC80914, ACC81440, ACC81547, ACC81570,
6, ACC83981, ACC84622, ACC84732, ACC85457      EKQ66830, EKQ66911, EKQ67039, EKQ67311, EKQ69997, EKQ70735
OG00000005      AFZ52318, AFZ52611, AFZ52613, AFZ52925, AFZ52973, AFZ53626, AFZ53840, AFZ53841, AFZ53849, AFZ
CC82559, ACC83603, ACC83674, ACC85005, ACC85009 EKQ67574, EKQ67809, EKQ69976
OG00000006      AFZ52319, AFZ53394, AFZ54017, AFZ54472      ACC79360, ACC79745, ACC79853, ACC80832, ACC80867, ACC
478, EKQ67551, EKQ67724, EKQ67810, EKQ68266
OG00000007      AFZ53704, AFZ54461, AFZ54462      ACC79786, ACC80242, ACC80282, ACC80538, ACC80768, ACC81387, A
2, EKQ68369, EKQ70142, EKQ70145, EKQ71300
```

The tool also generates [summary statistics](#) we can look through.

Could write a script to turn this into a table or use the [summary count table](#) provided.

ORTHOLOG_GRP	SP1	SP2	SP3
ORTHO_0001	10	5	0
ORTHO_0002	1	1	0