Orthologs and Paralogs

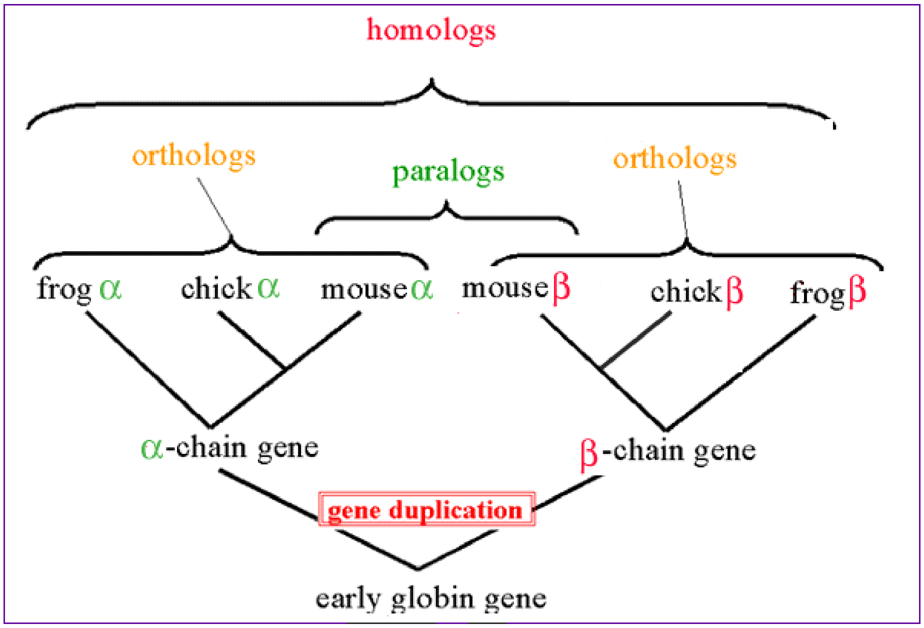
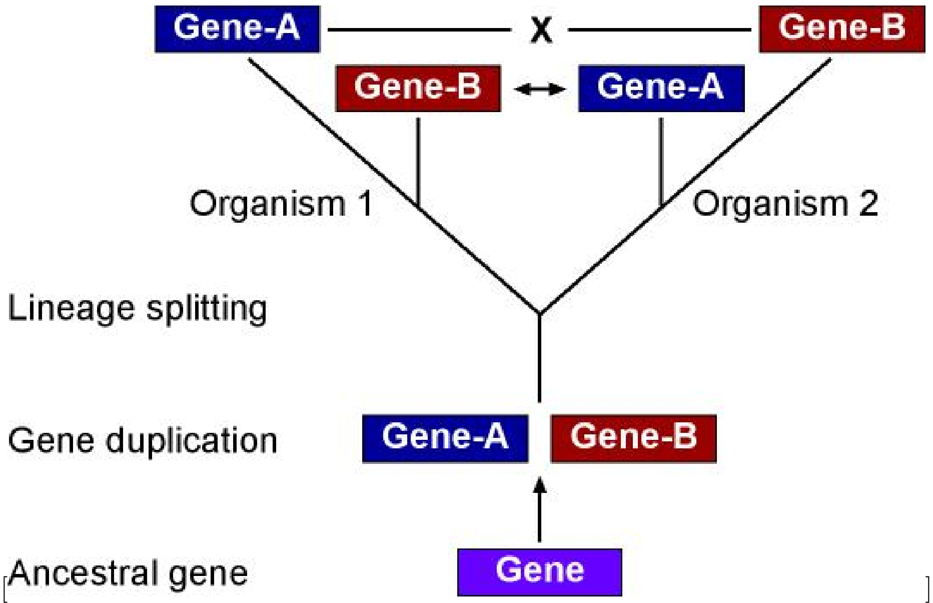


Figure 1: Orthologs

Gene families and Orthology

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

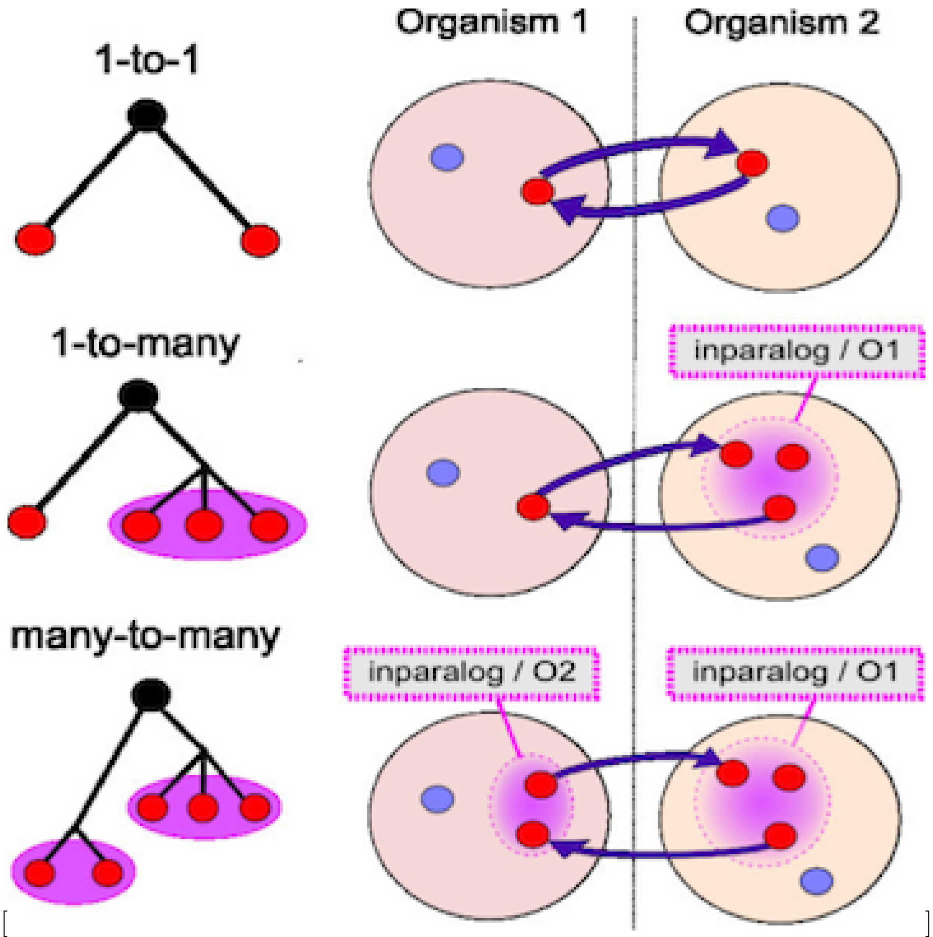
Genes can duplicate (Paralogs) and can be identical due to descent (Ortholog)

.center[

]

Methods

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

.center[]

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.

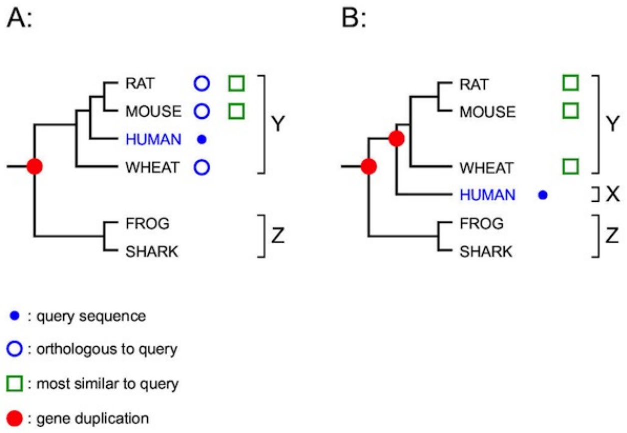


Figure 2: RIO

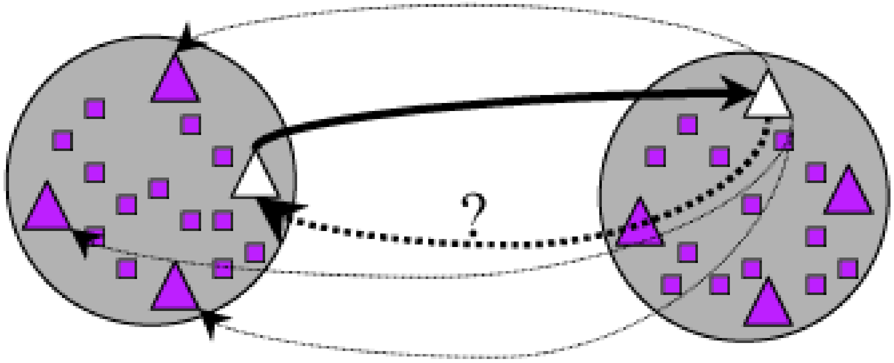


Figure 3: BRH

Let's walk through the code

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

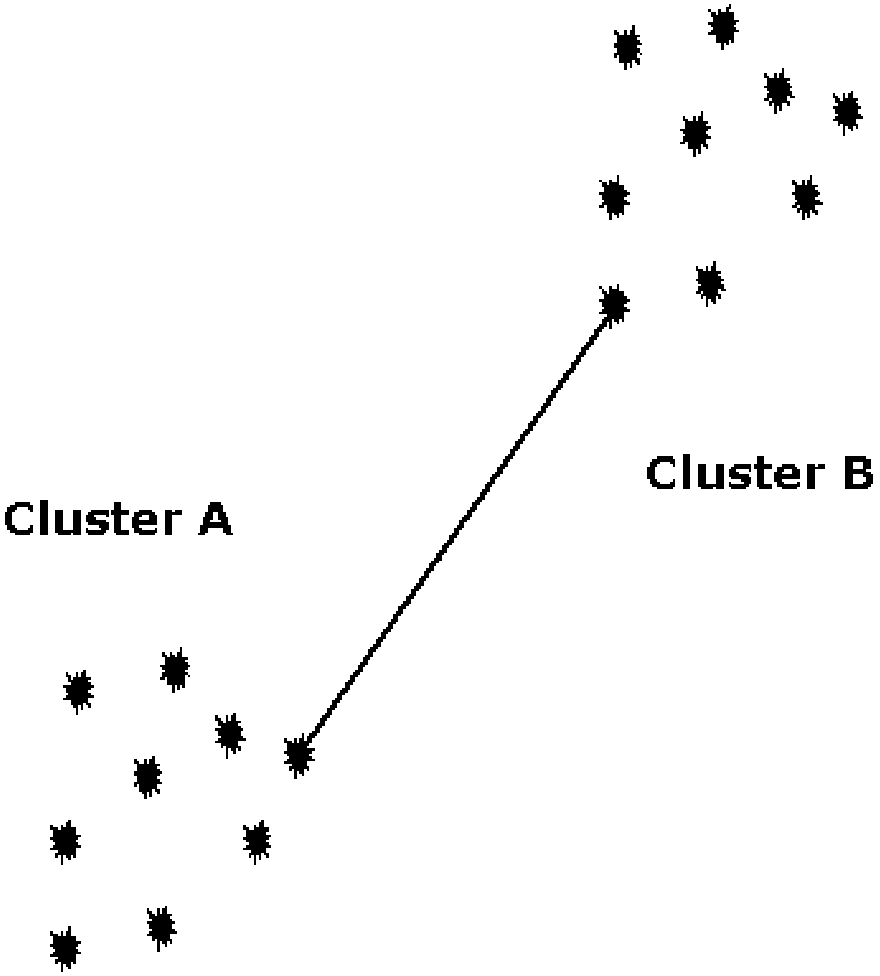


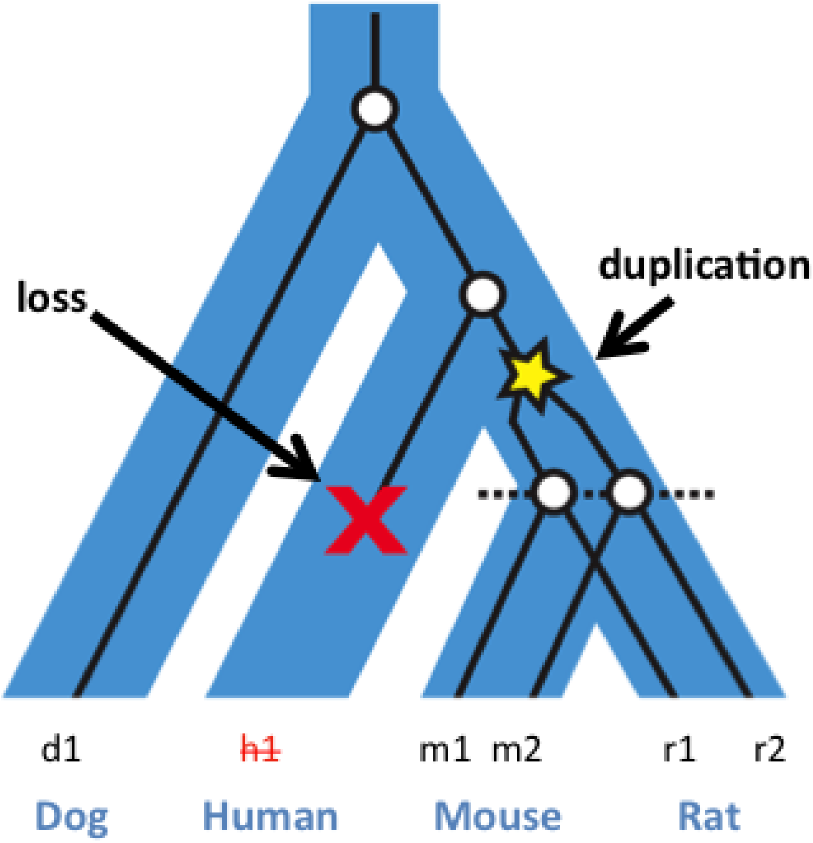
Figure 4: SingleLinkage

Code up single-linkage

Let's look at some code.

Will write this in Python in Class

Issues



.center[ ]

Existing solution

- OrthoMCL - requires SQL Database
- Orthagogue - nearly identical results but runs w/o DB

Steps to build orthologs on cluster

Make sure genome protein FASTA file is

```
>SPECIESPREFIX|GENENAME
```

See https://github.com/biodataprog/GEN220_2019_examples/tree/master/Bioinformatics_1 for example script for running orthofinder.

```
#!/usr/bin/bash
#SBATCH --ntasks 16 --mem 8G -p short
module load ncbi-blast
module load orthofinder
module load miniconda2
CPU=8

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

for file in *.fa.gz
do
 m=$(basename $file .pep.all.fa.gz)
 pigz -dc $file > $m.fasta
done
pigz -k *.fa.gz
cd ..

orthofinder.py -a $CPU -f cyanobacteria
```

Ortholog results

Opening the file cyanobacteria/Results_Nov08/Orthogroups.txt

Format

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

```
    Cyanobacterium_aponinum_pcc_10605.ASM31767v1    Nostoc_punctiforme_pcc_73102.ASM2002v1
OG0000000           EKQ66605, EKQ66611, EKQ66662, EKQ66782, EKQ66954, EKQ66984, EKQ67084, EK
 EKQ67433, EKQ67590, EKQ67680, EKQ67799, EKQ67807, EKQ67983, EKQ68026, EKQ68032, EKQ68054, E
EKQ69279, EKQ69300, EKQ69345, EKQ69368, EKQ69506, EKQ69549, EKQ69629, EKQ69630, EKQ69655, EK
```

```
KQ70786, EKQ70840, EKQ70870, EKQ70894, EKQ71088, EKQ71090, EKQ71265, EKQ71335
OG0000001   AFZ52442, AFZ54265, AFZ54640    ACC78968, ACC78978, ACC79054, ACC79090, ACC79138
81797, ACC82091, ACC82628, ACC82978, ACC83035, ACC83215, ACC83711, ACC84528, ACC84844, ACC84
69971, EKQ69995, EKQ70003, EKQ70556, EKQ70833, EKQ71286
OG0000002   AFZ55137    ACC79344, ACC80485, ACC80595, ACC82143, ACC82836, ACC82962, ACC83845
 ACC84972, ACC84974, ACC84981, ACC84982, ACC84983, ACC85032 EKQ66950, EKQ67597, EKQ67615, EK
OG0000003   AFZ53198    ACC78875, ACC78976, ACC79256, ACC79524, ACC79759, ACC80145, ACC80528
 ACC82769, ACC83025, ACC83081, ACC83457, ACC83602, ACC83721, ACC83749, ACC84422, ACC85331
OG0000004       ACC80422, ACC80525, ACC80662, ACC80851, ACC80857, ACC80914, ACC81440, ACC815
6, ACC83981, ACC84622, ACC84732, ACC85457   EKQ66830, EKQ66911, EKQ67039, EKQ67311, EKQ69997
OG0000005   AFZ52318, AFZ52611, AFZ52613, AFZ52925, AFZ52973, AFZ53626, AFZ53840, AFZ53841,
CC82559, ACC83603, ACC83674, ACC85005, ACC85009 EKQ67574, EKQ67809, EKQ69976
OG0000006   AFZ52319, AFZ53394, AFZ54017, AFZ54472  ACC79360, ACC79745, ACC79853, ACC80832,
478, EKQ67551, EKQ67724, EKQ67810, EKQ68266
OG0000007   AFZ53704, AFZ54461, AFZ54462    ACC79786, ACC80242, ACC80282, ACC80538, ACC80768
2, EKQ68369, EKQ70142, EKQ70145, EKQ71300
```

Write script to turn this into a table

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