Practical Bioinformatics

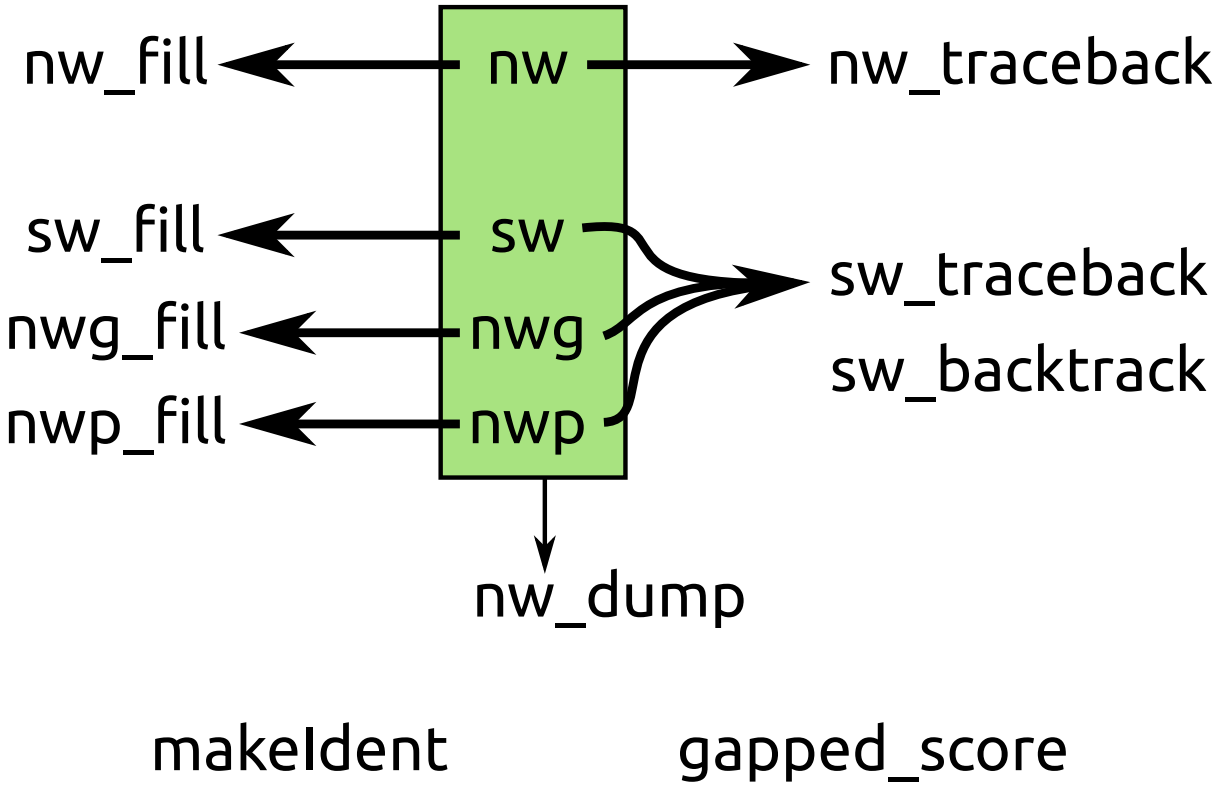
Mark Voorhies

5/9/2017

Habits are things you get for free, without requiring any special work.

–Cory Doctorow *Advice to Writers, 4/5/2012*

Functions in homework solution (dp2.py)



Needleman-Wunsch with g=0

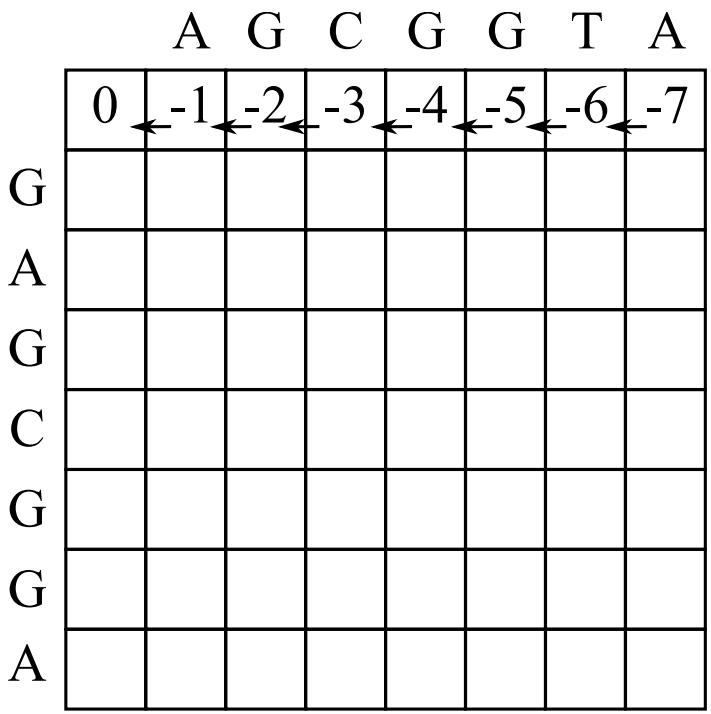
		A	G	C	G	G	T	A
G								
A								
G								
C								
G								
G								
A								

Needleman-Wunsch with g=0

		A	G	C	G	G	T	A
	0							
G								
A								
G								
C								
G								
G								
A								

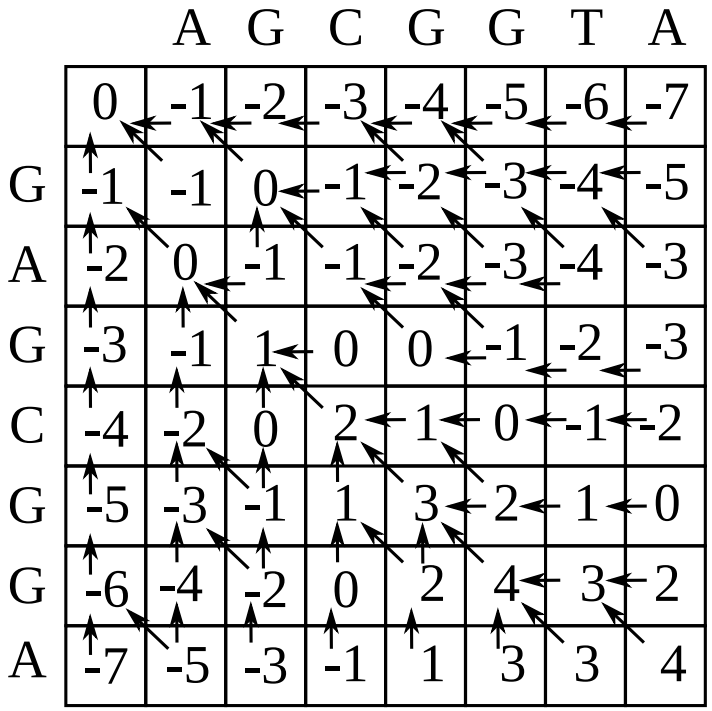
```
def nw_fill(seq1, seq2, s, e):
    # Initialize dp matrix
    #     first dimension  = seq1 positions
    #     second dimension = seq2 positions
    #     m[i][j] = best score for subalignment
    #                   of seq1[:i],seq2[:j]

    m = [[0]]
    # Initialize pointer matrix, a two dimensional
    #   matrix of lists of (row, column) pointers
    p = [[None]]
```

Needleman-Wunsch with g=0



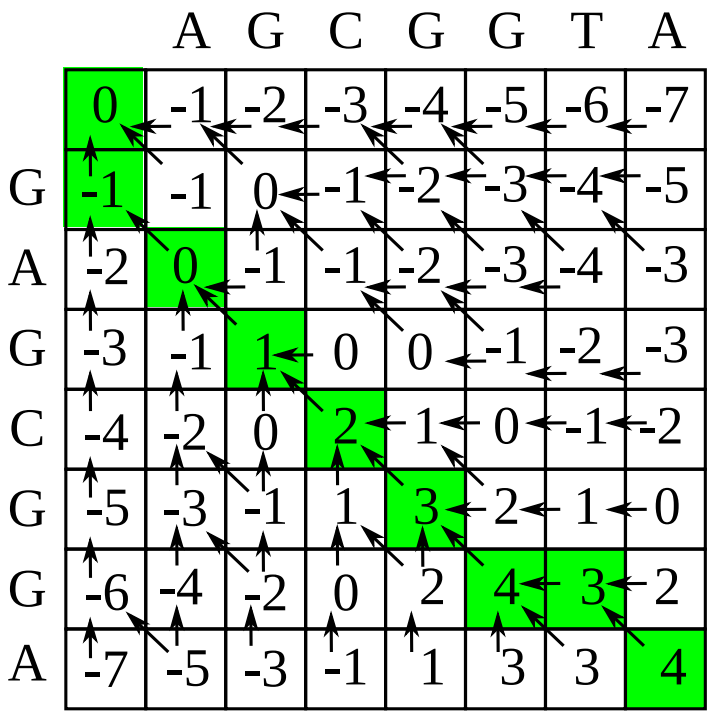

```
# Fill first row as leading gaps
for j in range(len(seq2)):
    m[-1].append(m[0][j]+e)
    p[-1].append([(0,j)])
```

Needleman-Wunsch with g=0



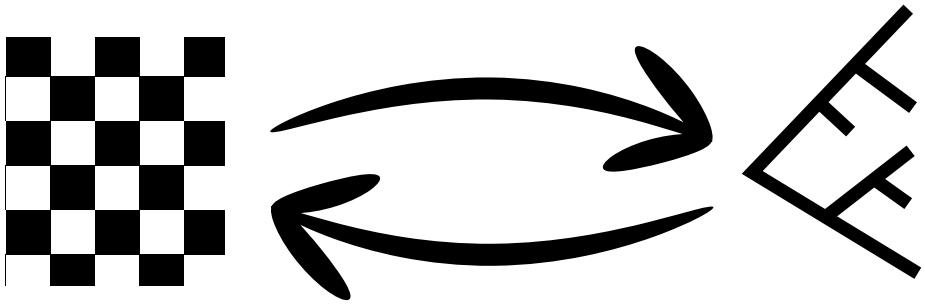

```
for i in range(len(seq1)):
    # First column is leading gaps
    m.append([m[i][0]+e])
    p.append([[(i,0)]])
    for j in range(len(seq2)):
        # Score for aligning seq1[i] with seq2[j]
        match = m[i][j]+s[seq1[i]][seq2[j]]
        # Score for aligning seq1[i] with a gap
        hgap = m[i+1][j]+e
        # Score for aligning seq2[i] with a gap
        vgap = m[i][j+1]+e

        best = max(match, vgap, hgap)
        m[-1].append(best)
        p[-1].append([])

        if(match == best):
            p[-1][-1].append((i,j))
        if(hgap == best):
            p[-1][-1].append((i+1,j))
        if(vgap == best):
            p[-1][-1].append((i,j+1))
```

Needleman-Wunsch with g=0




```
# Start at bottom right corner
curpos = (len(seq1),len(seq2))
aligned1 = ""
aligned2 = ""

exitFlag = False
for i in range(len(seq1)+len(seq2)):
    plist = p[curpos[0]][curpos[1]]
    if(plist is None):
        exitFlag = True
        break
    nextpos = plist[0]
    # Check for vgap
    if(nextpos[0] == curpos[0]):
        aligned1 = "-"+aligned1
    else:
        aligned1 = seq1[nextpos[0]]+aligned1
    # Check for hgap
    if(nextpos[1] == curpos[1]):
        aligned2 = "-"+aligned2
    else:
        aligned2 = seq2[nextpos[1]]+aligned2

    curpos = nextpos

if(exitFlag == False):
    print "WARNING: Unexpected exit from traceback"
```

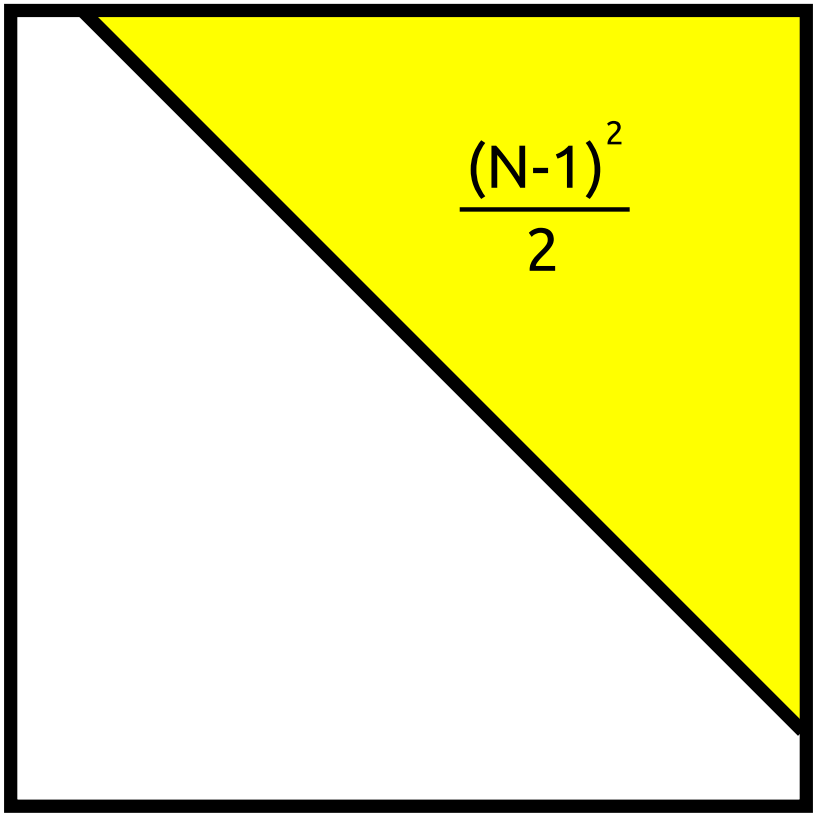
Evolution implies a self-consistent model



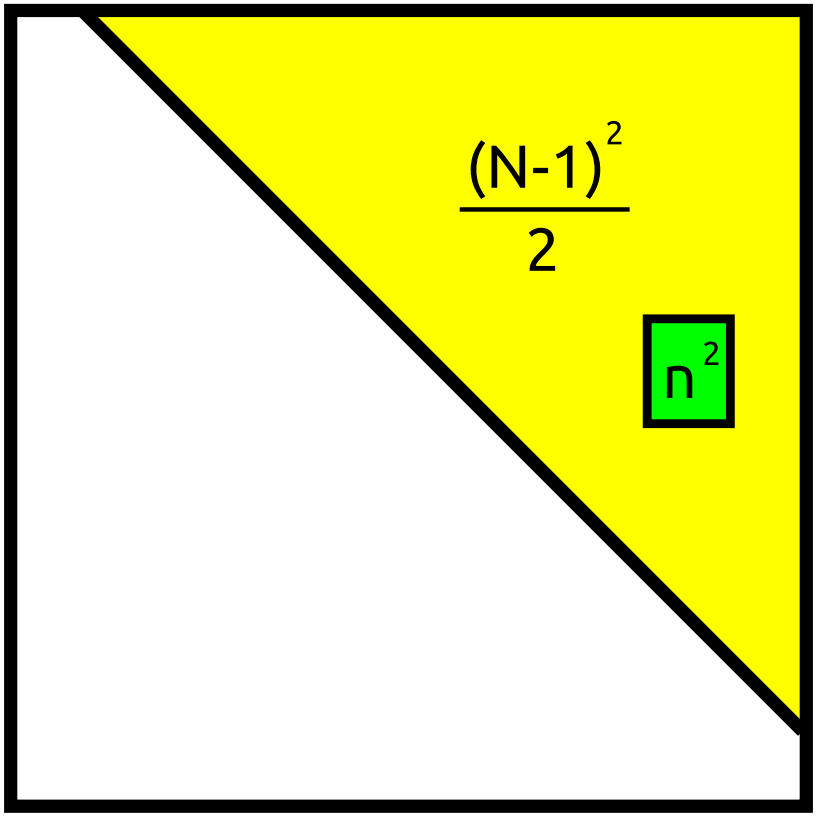
Distances
(Pairwise relationships)

Topology
(Evolutionary history)

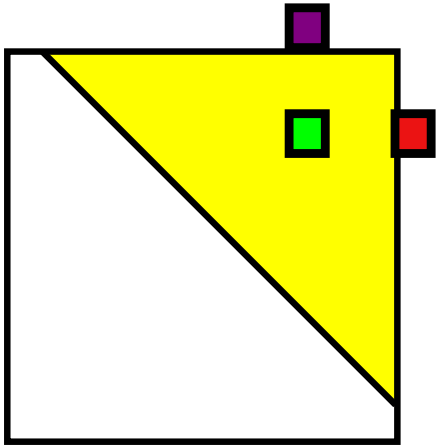
Measure all pairwise distances by dynamic programming

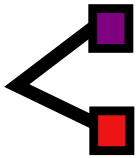


Measure all pairwise distances by dynamic programming

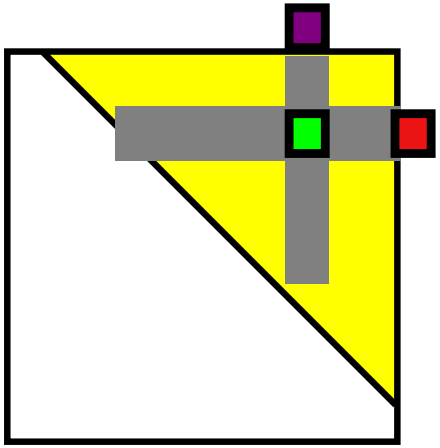


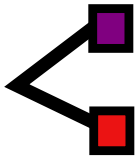
Generate a guide tree by UPGMA



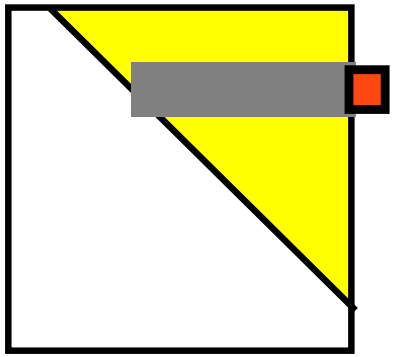


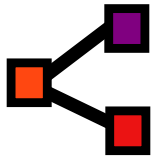
Generate a guide tree by UPGMA



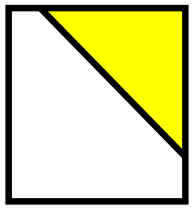


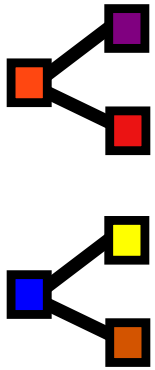
Generate a guide tree by UPGMA



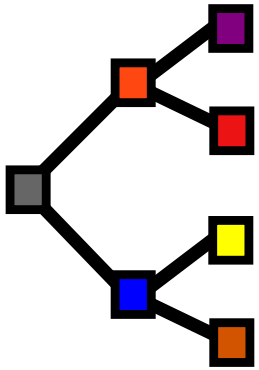


Generate a guide tree by UPGMA

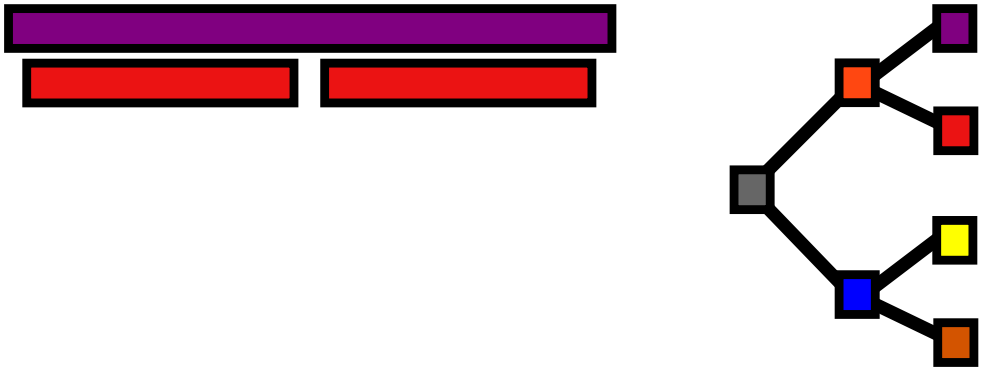




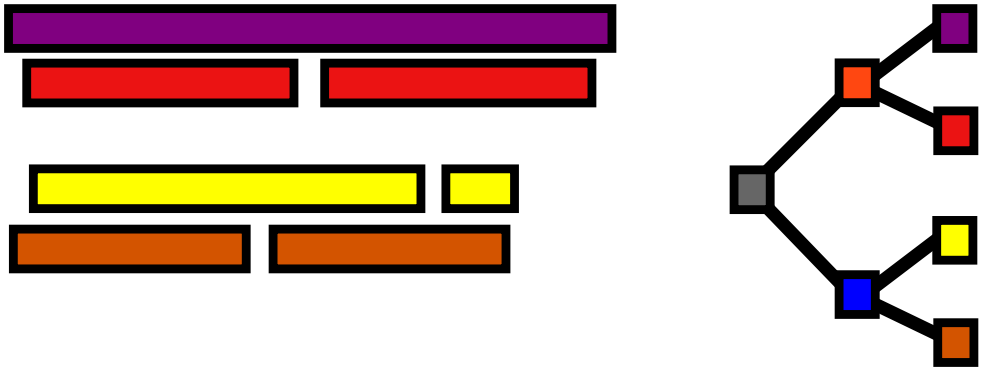
Generate a guide tree by UPGMA



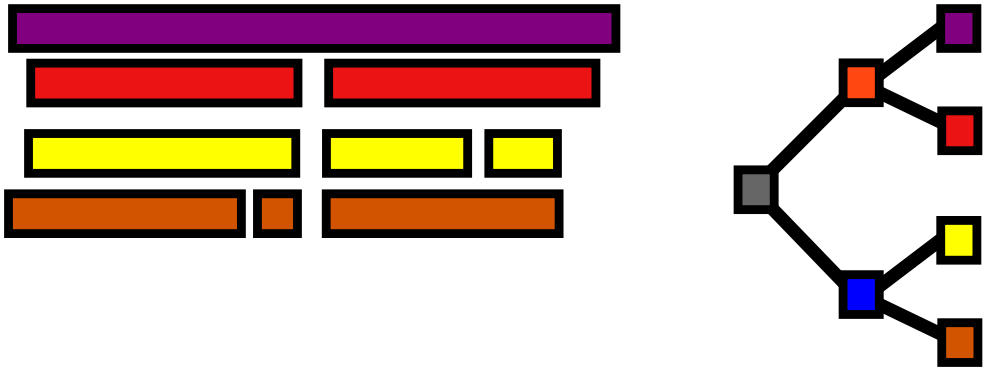
Progressive alignment following the guide tree



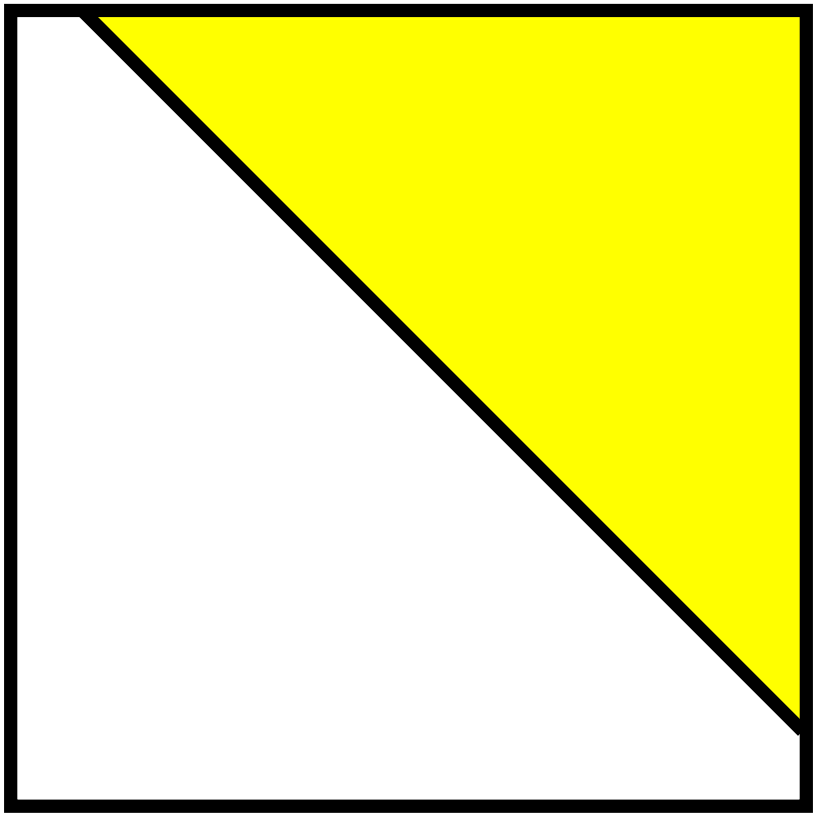
Progressive alignment following the guide tree



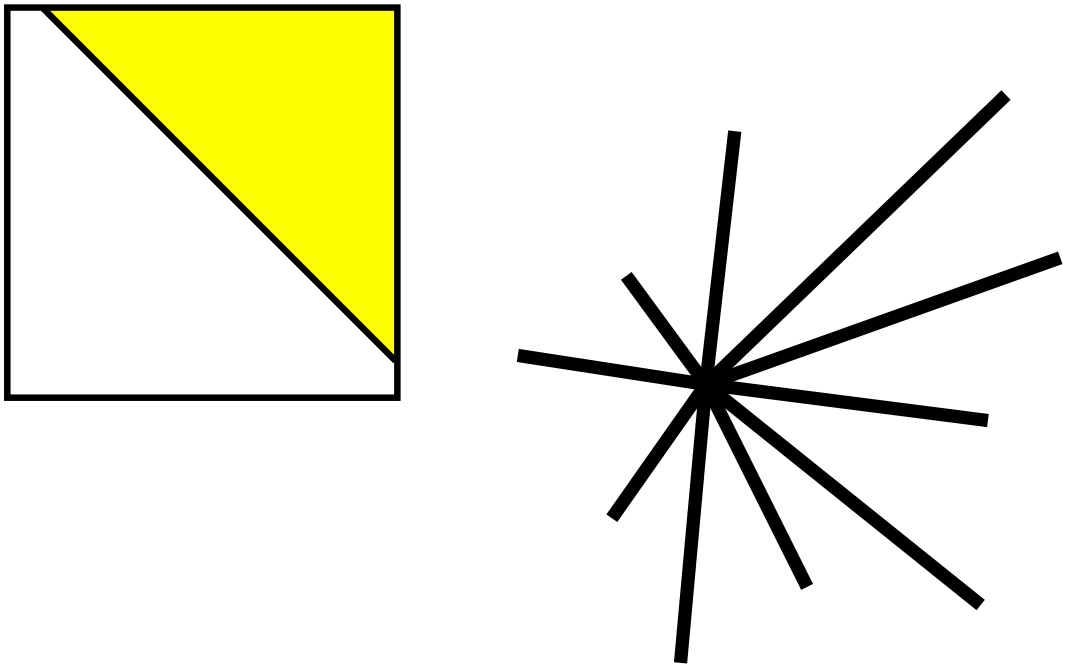
Progressive alignment following the guide tree



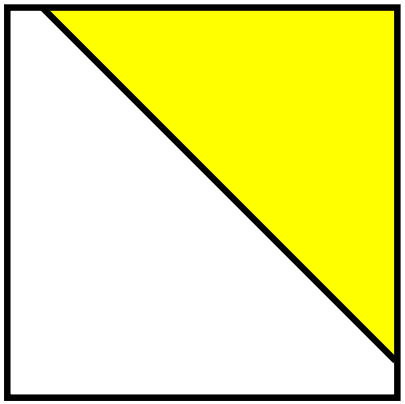
Measure distances directly from the alignment

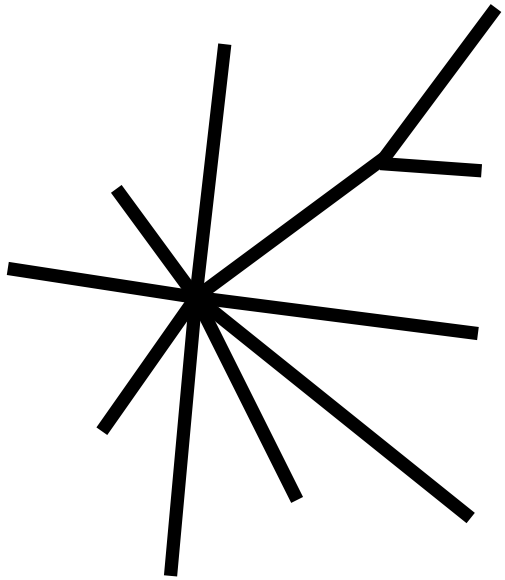


Generate neighbor-joining tree from new distances

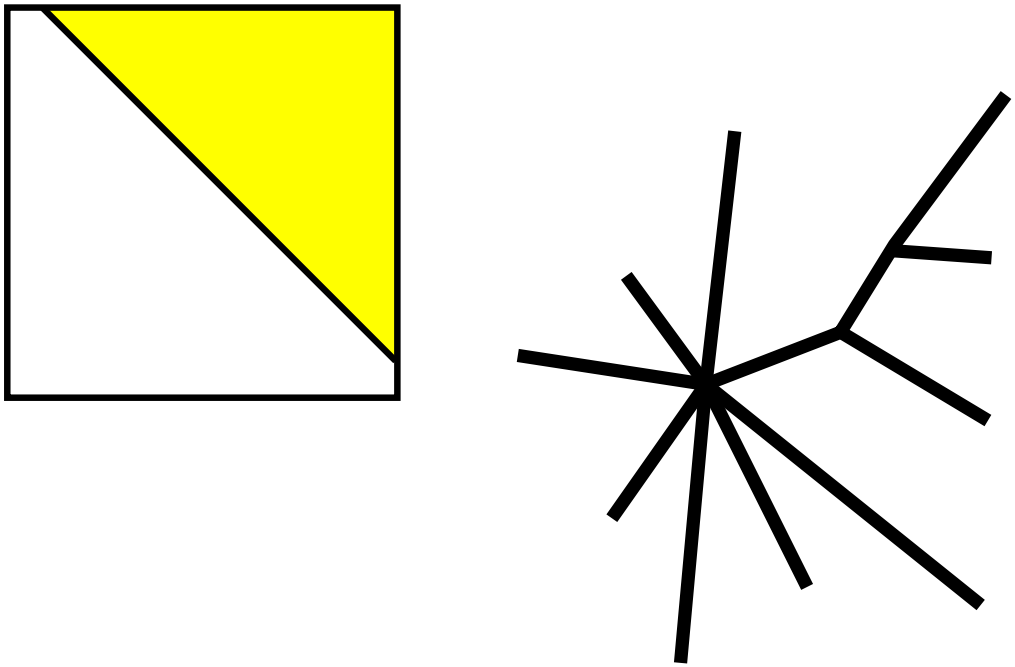


Generate neighbor-joining tree from new distances

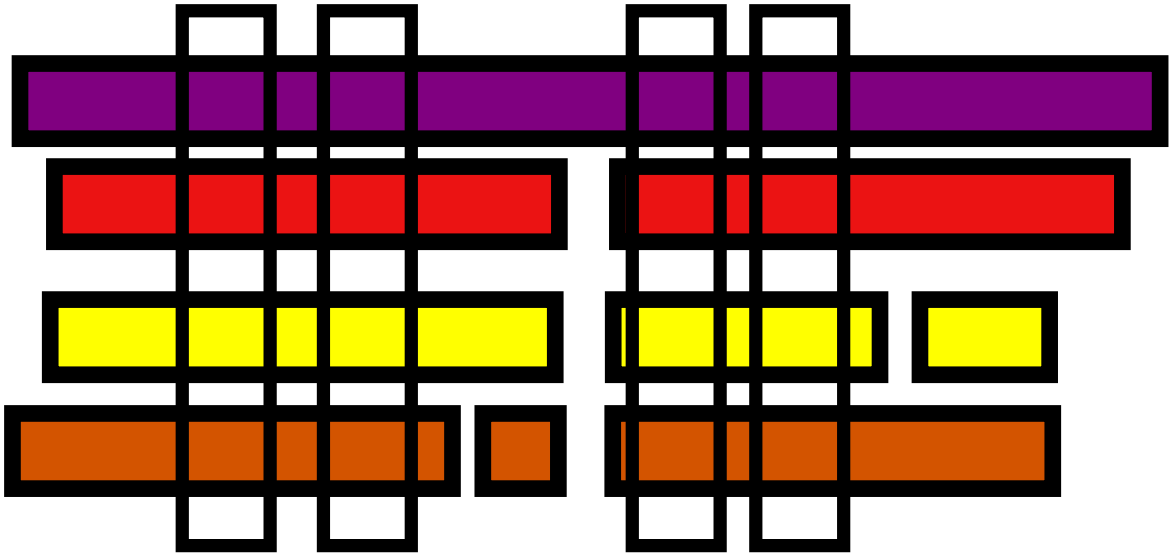




Generate neighbor-joining tree from new distances

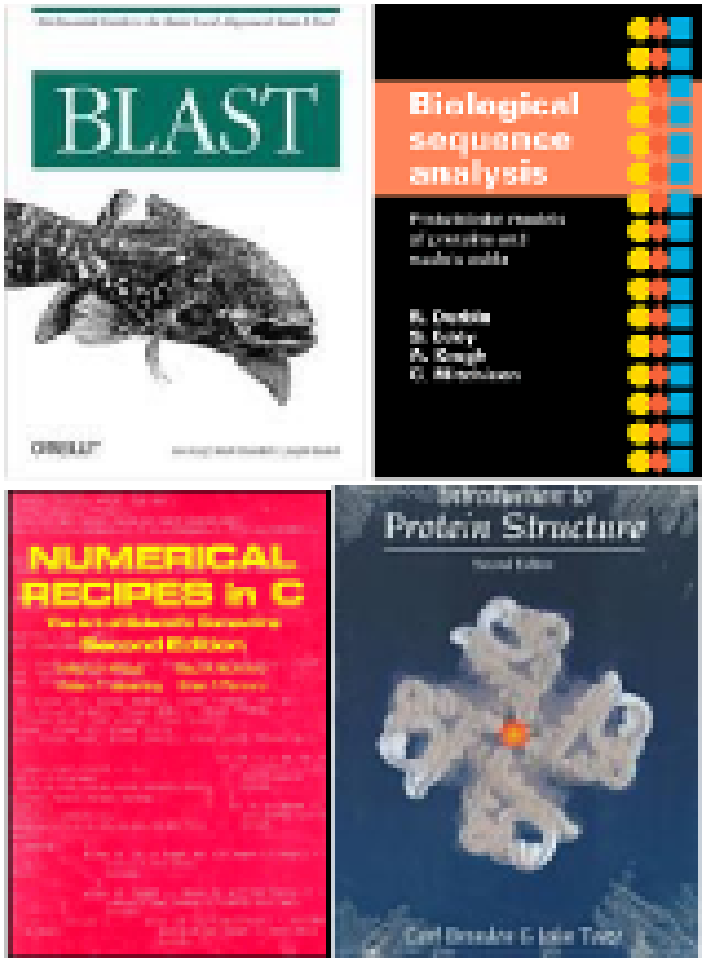


Generate bootstrap values from subsets of the alignment



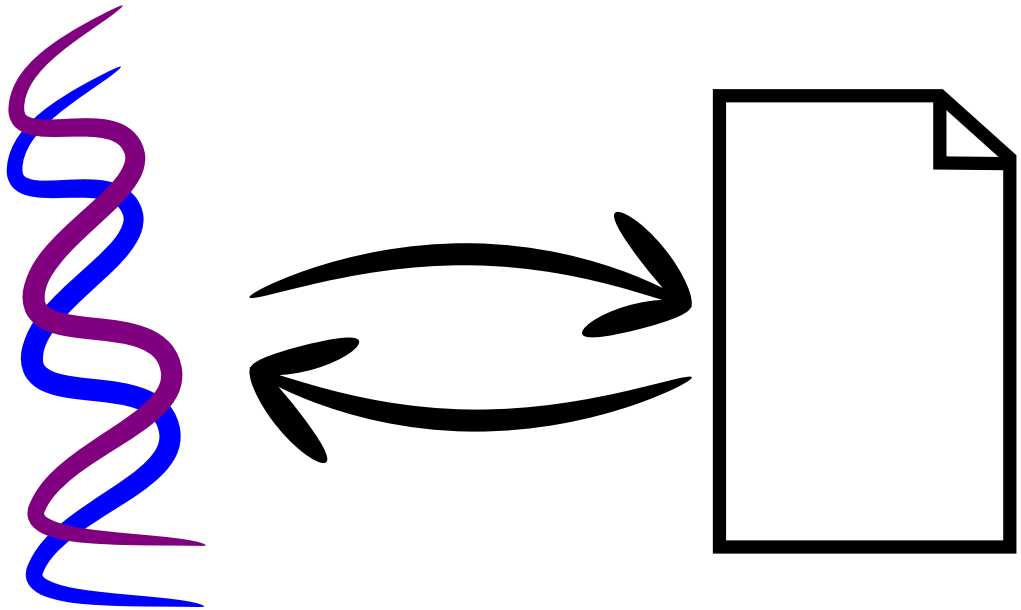
Viewing the alignment and tree in JALVIEW

RTFM



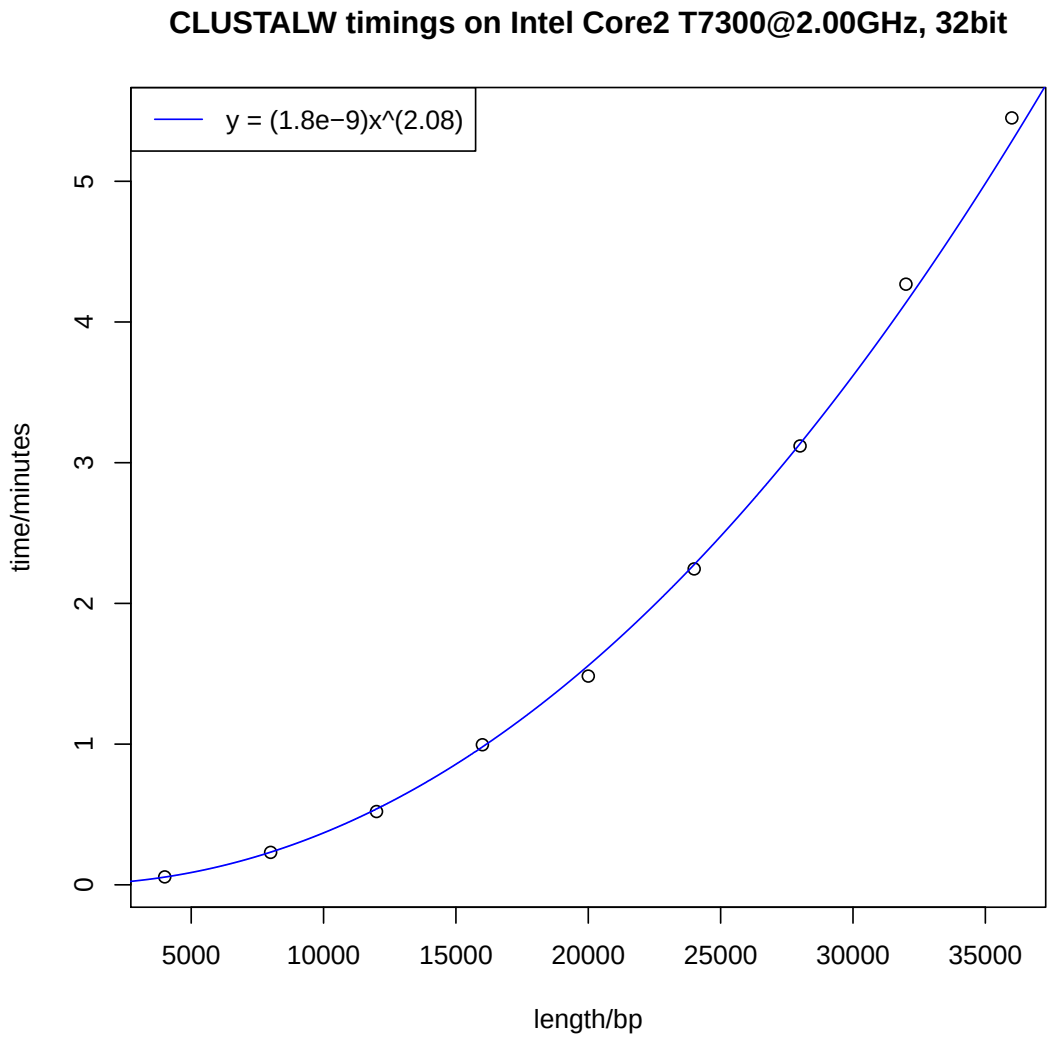
- For tools:
  - Read the manual
  - Read the paper
- Good general references:
  - The O'Reilly BLAST book
  - Durbin, Eddy, Krogh, and Mitcheson (HMMs)
  - Numerical Recipes
  - Branden & Tooze

Every object should have an isomorphism to a file

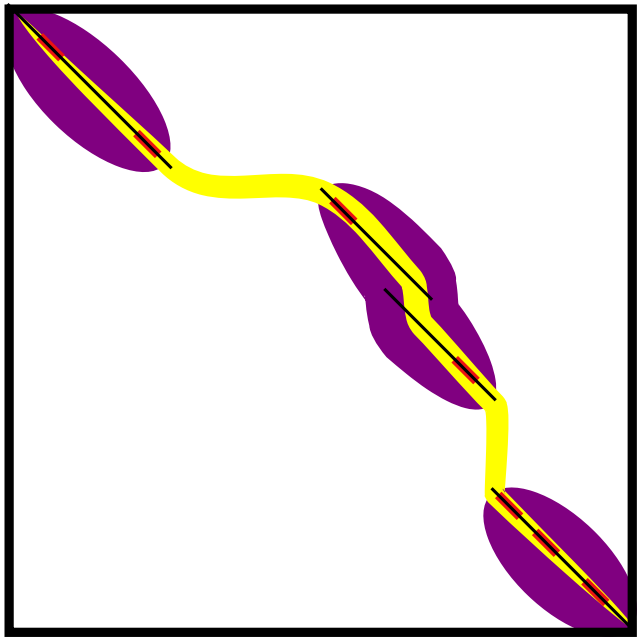


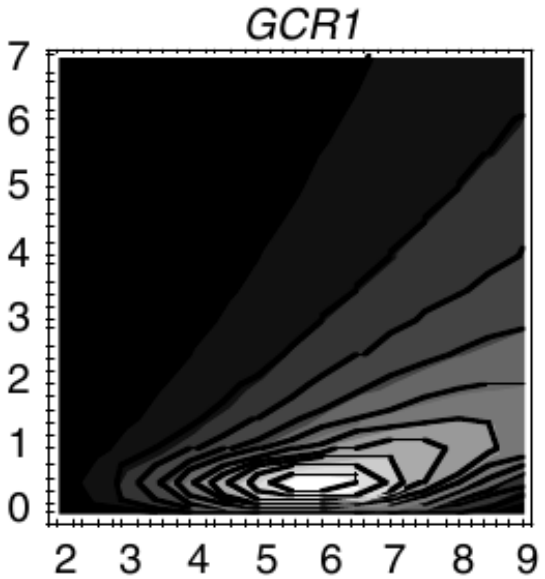
- Export, audit, edit, and import *independent* of a given program.
- Standard file formats for portability.
- Don't be afraid to look inside and hack on *your* data files.

Run times are predictable and measurable



Heuristics and stochastic sampling for hard problems



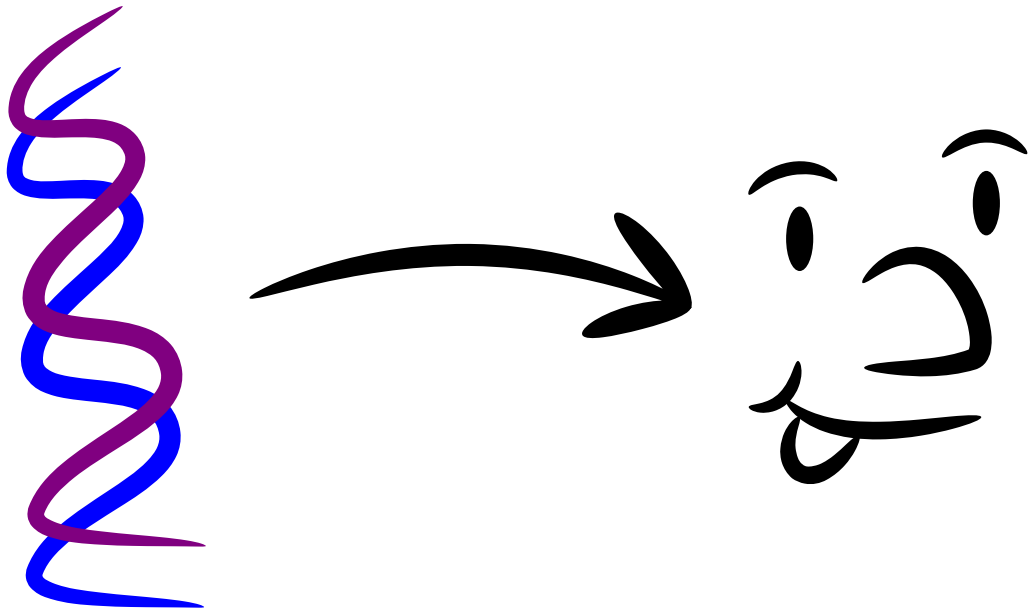


BLAST, HMMer3, simulated annealing, ...

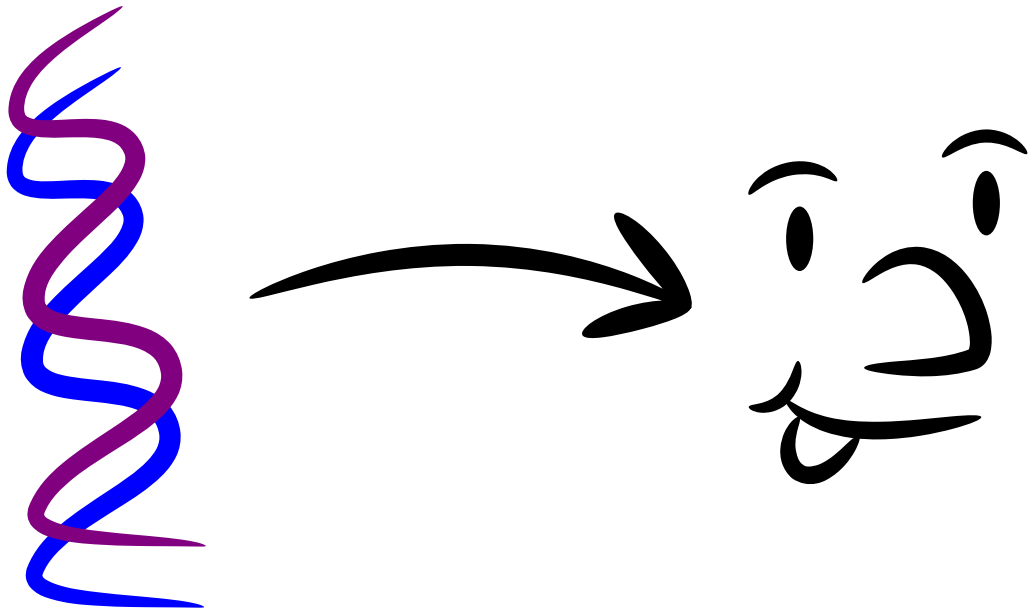
Evolution is a rich source of information

- Infer homology from sequence similarity
- More sequences provide more information

Phenotype is more diverse than Genotype

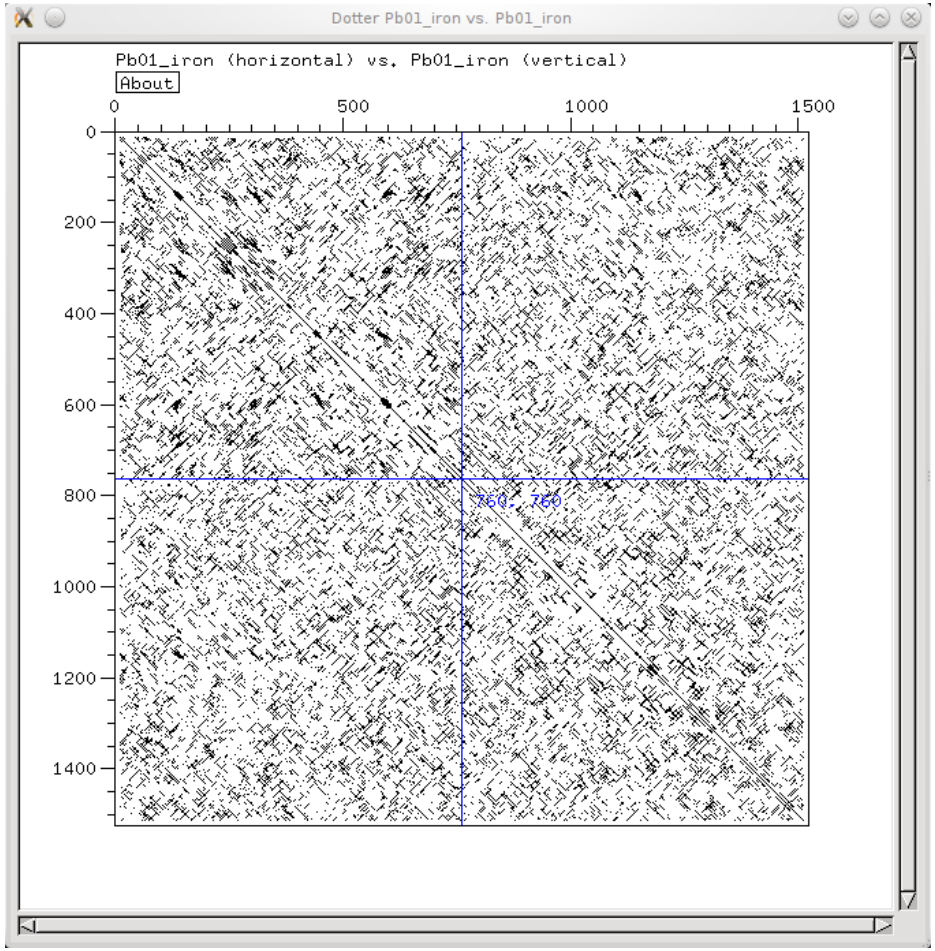


Phenotype is more diverse than Genotype

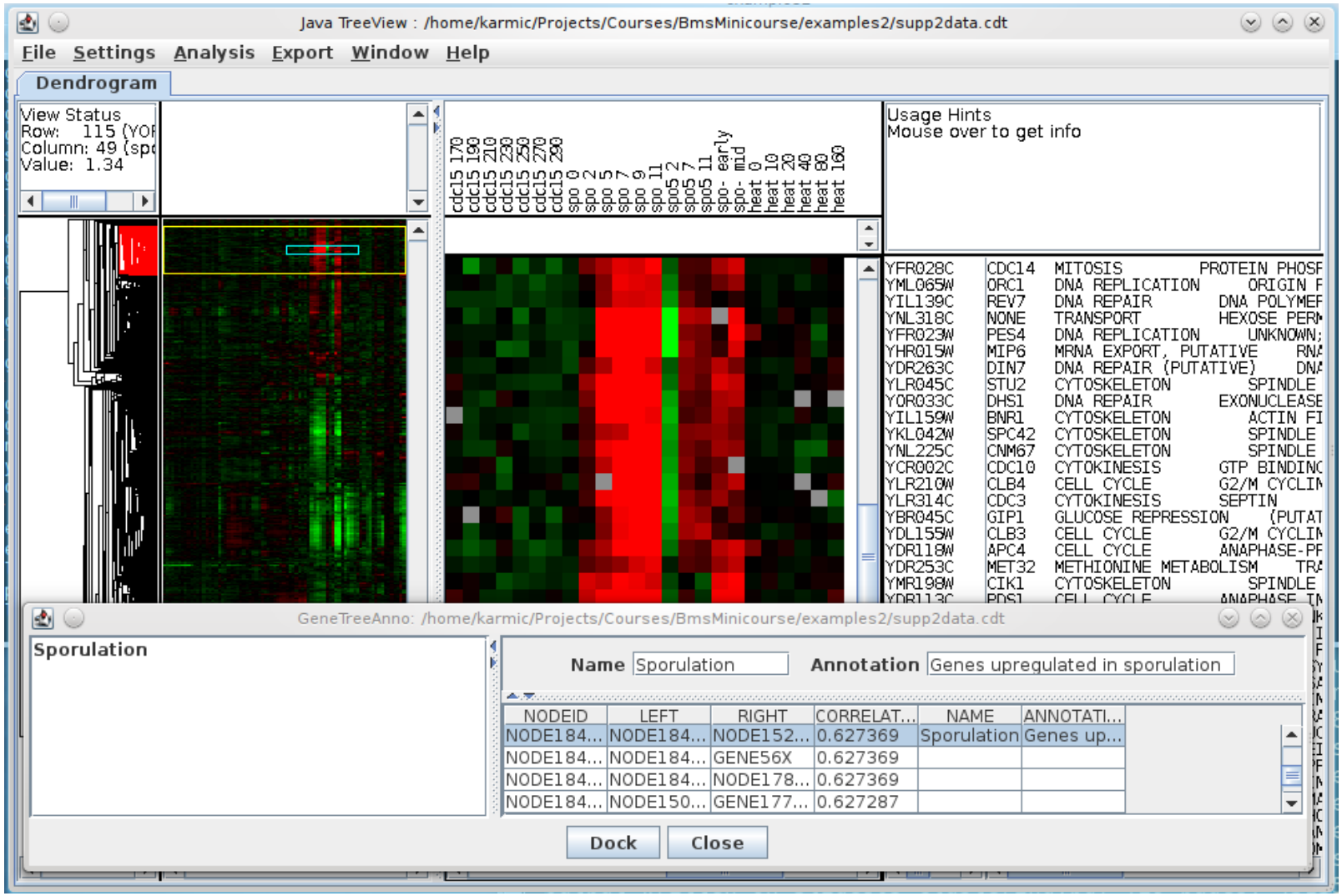


- Make sure you know what you are measuring
- Nucleic acid sequences are especially easy to address
- Many phenotypes can be analyzed by common numerical methods

Start from an unbiased view



Tools should support aggregation and annotation



Science is a Conversation

- Follow computational methods as they evolve (Web of Science; RSS: PubMed, GEO, arXiv, ...)

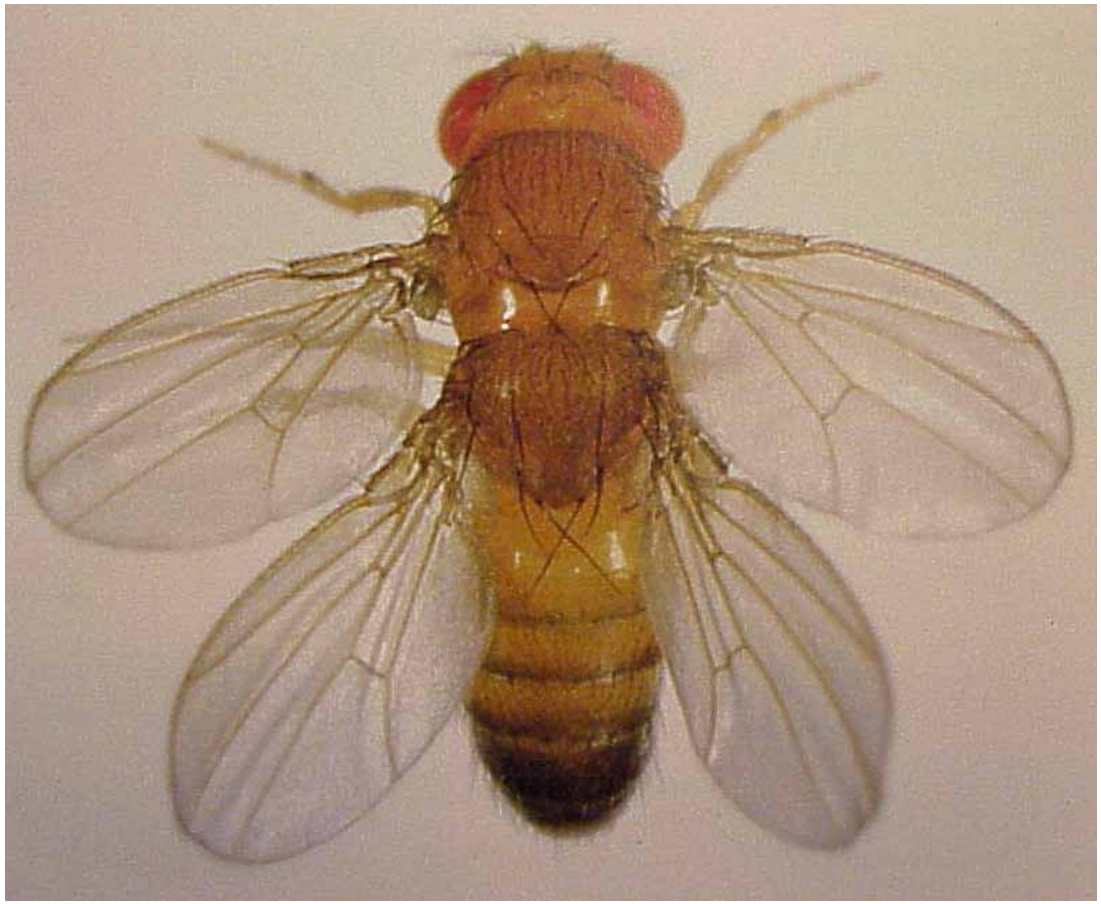
Science is a Conversation

- Follow computational methods as they evolve (Web of Science; RSS: PubMed, GEO, arXiv, ...)
- As a reviewer, insist on availability of source code

Science is a Conversation

- Follow computational methods as they evolve (Web of Science; RSS: PubMed, GEO, arXiv, ...)
- As a reviewer, insist on availability of source code
- Draw on your classmates' expertise

We understand systems by breaking them



Source: Peter A. Lawrence via http://www.bio.davidson.edu/courses/molbio/ubx/ubx.html