Homology and Pairwise Alignment

James B. Pease[1] and Stephen A. Smith[2]

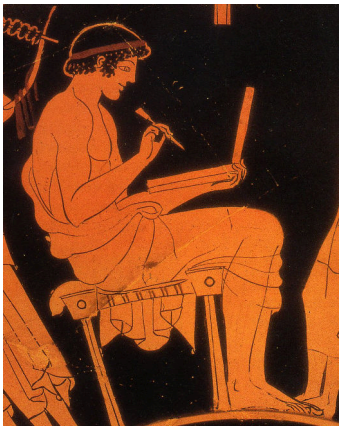
[1]Department of Biology, Wake Forest University
[2]Department of Ecology and Evolutionary Biology, University of Michigan

8 May 2018

1 By the end of this tutorial, your goal is to be able to:

- Understand the relationship between sequence data and hypothesis testing in molecular evolution.
- Describe homology, orthology, and paralogy in the context of molecular evolution.
- Explain the difference between sequence similarity and homology.
- Use pairwise alignment as a tool for the inference of sequence homology.
- Understand the BLAST algorithm, interface, features, and potential pitfalls.

2 The Art and Science of Computational Biology



Ὁ βίος βραχύς,
ἡ δὲ τέχνη μακρή,
ὁ δὲ καιρὸς ὀξύς,
ἡ δὲ πεῖρα σφαλερή,
ἡ δὲ κρίσις χαλεπή.
— Ἱπποκράτης, Αφ. 1.1

Life is short,
and the art long,
opportunity fleeting,
experience perilous,
and decision difficult.
— Hippokrates, Aph. 1.1

Mastering quantitative evolutionary biology takes time,
is often limited by the available data,
requires much learning by trial-and-error,
and developing an intuition for complex decisions.

3 An Example of a Basic Molecular Evolution Workflow

1. **Formulate a testable molecular evolution question**
2. **Collect sequence data**
3. **Determine sequence homology**
4. *Align homologous sequence data*
5. *Make a ortholog/paralog/homolog tree (usually)*
6. *Analyze the tree and/or sequence data to model evolutionary relationships, evolutionary events, or variation in evolutionary processes.*

4 What is homology?

4.1 Phenotypic Homology

Phenotypic traits shared among a set of organisms are called **homologous** when all organisms inherit the trait from a common ancestor that also had the given trait. Here "trait" can mean *any* property of an organism. The leaves of the olive tree and kermes oak in Figure 1 are obviously homologous to each other because the ancestor of flowering plants also had leaves derived from the same tissues. The spines of cacti (from the family *Cactaceae*) are also homologous to these leaves, because they are modified leaves and thus derived from the same ancestral tissue.

By contrast, an **analogous trait** is shared among organisms, but the ancestor does not have the trait. This means the trait developed separately in each lineage. For example, euphorbs (from the family *Euphorbiaceae*) also have pointy thorns, but these spines are protrusions of stem tissue, and so are analogous (but NOT homologous) to cactus spines. The starchy tuber of the sweet potato is a modified root, while the starchy tuber of the potato is a modified stem (Fig. 2). Therefore, these structures are analogous, because the ancestor did not have tubers. So traits can be "similar" because they are homologous shared feature (inherited from the ancestor) or just due to separate processes of adaptation that led to the development of analogous features.

Figure 1: The leaves of an olive tree (*Olea europaea ssp. europaea*), kermes oak (*Quercus coccifera*), and the spines of prickly pear and Greek Spiny Spurge (*Euphorbia acanthothamnos*).

Figure 2: Tubers of sweet potato (*Ipomoea batatas*) and potato (*Solanum tuberosum*).

4.1.1 ✎ Is egg-laying in platypus and chicken analogous or homologous?

4.2 Molecular Homology

In addition to more commonplace physiological traits, molecular traits can also be homologous, including the structures of chromatin, RNA, and proteins, pathways, and expression profiles. For example, core developmental pathways are often conserved among even distantly related organisms (Figure 3). The genome-wide patterns of expression can also show homology (Figure 4) if organisms maintain similar expression profiles to a shared ancestor.

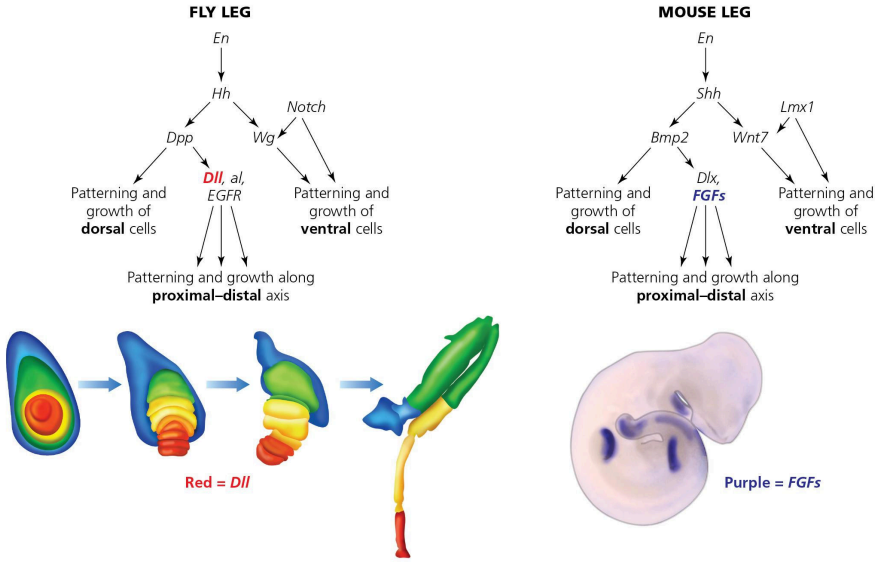


Figure 3: Molecular signaling pathways for fly (*left*) and mouse (*right*) leg development.

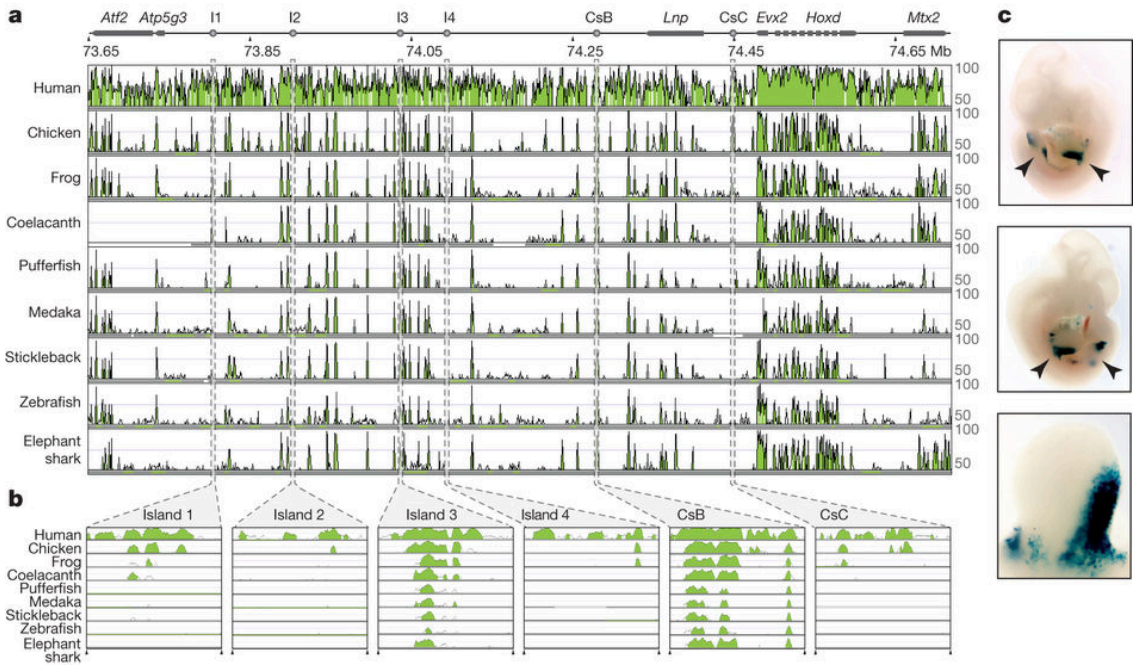


Figure 4: Expression levels of regulatory forelimb genes across several tetrapods shows homologous patterns of expression. (Amemiya, et al. 2013 *Nature*.)

4.3 Sequence Homology

Homology can also be characterized at the sequence level, and this is the type of homology that most often is used molecular evolution. **In biological sequences, the "traits" are the sequence character states** for either nucleotides (`ATGC`) **or amino acids** (`ACDEFGHIKLMNPQRSTVWY`.

Sequence homology occurs when organisms have the same molecular sequence characters because they all inherited these characters from their shared ancestor. For example, the highly conserved NFκB protein sequence (Fig. 5) shows strong evidence of homology of the amino acid character states (i.e., "the letters of the code").

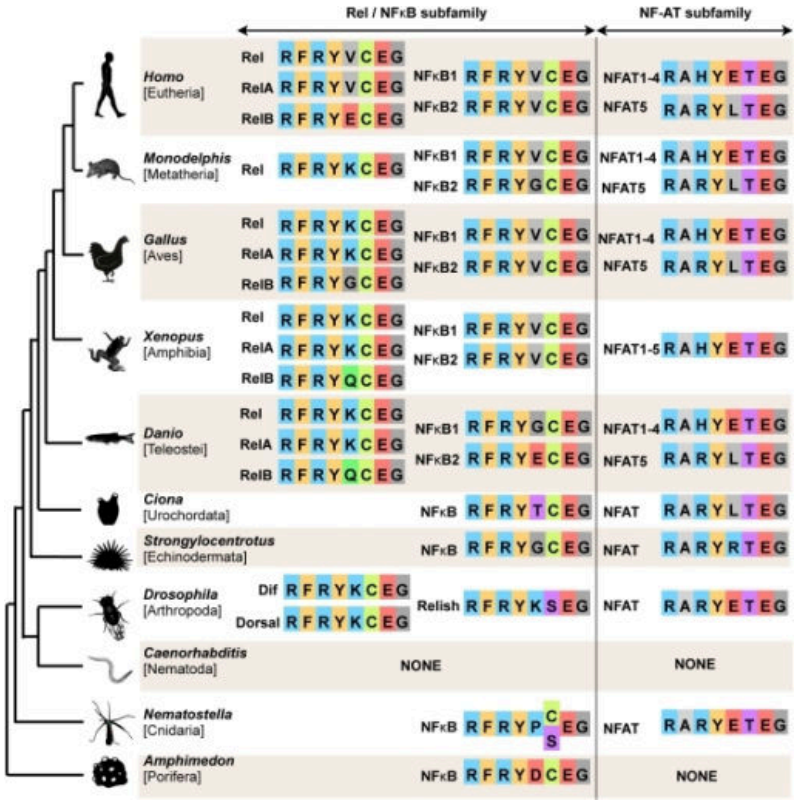


Figure 5: A phylogeny of NFκB showing several conserved and some variable amino acids (Sullivan, et al. 2009 *PLoS ONE*).

4.4 Why is homology important for studying molecular evolution?

4.4.1 ✎ Your colleague comes to you and tells you that they sequenced the human *BRCA1* gene, *RHO* from chimpanzee, and *CFTR* from gorilla. Formally describe why a molecular evolution analysis might be difficult.

The above scenario is purposefully absurd to illustrate a foundational concept in molecular evolution. Molecular evolutionary questions (generally) test the **variation in evolutionary processes** among populations using **homologous sequence data**. In our hypothesis testing framework, this means the gene's "identity" is considered an experimental "constant" since homologous sequences have a shared point of origin. The evolutionary processes are the "variable" to be studied, as they differentially affect the ancestral sequence as it changes and diversifies in different populations.

4.5 Orthology and Paralogy

- **Homologous**: descended from a common ancestor
- **Paralogous**: descended from a common ancestor and split by a gene duplication event
- **Orthologous**: descended from a common ancestor and split by a speciation event

In Figure 6, all four genes are homologous because they all descend from common ancestral gene (red line at the bottom). The "green" and red "genes" are **paralogous genes** or **"paralogs,"** both in the present species and the ancestral species. After the speciation of species "F" and "H," the copies of the green gene in F and H are **orthologous genes** or **"orthologs"** because they are separated by a speciation event (but not a duplication event).

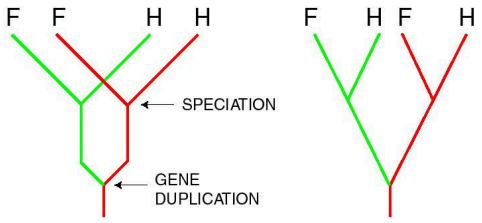


Figure 6: Phylogeny showing an ancestral gene (red line at the bottom) that is duplicated by a gene duplication event into "red" and "green" copies. Subsequently this ancestral population splits into two species: F and H. Species F and H both have a copy of the red and green gene.

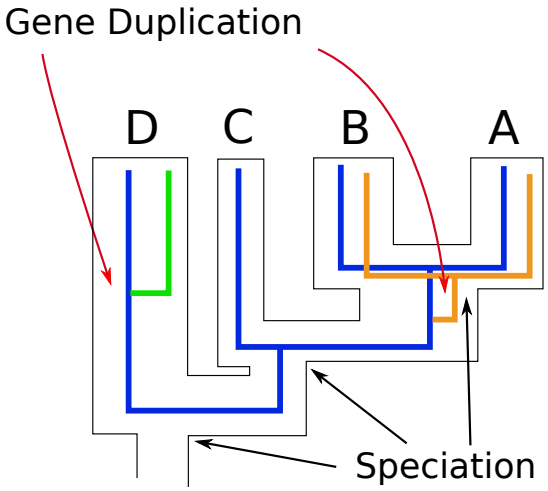


Figure 7: Another representation of orthologs vs. paralogs. Species are A, B, C, and D, and gene duplicates are represented by colors.

4.5.1 ✎ In Figure 7, what sequences are paralogous to D-blue? Orthologous to A-orange?

4.6 Should I collect data from orthologs or paralogs?

That depends on your question. Remember we want to study **variation in evolutionary processes** using **homologous molecular sequences** as the data. If you want to study **variation among duplicated sequences**, then use paralogs. However, most often we want to study **variation among species**, so orthologs are the appropriate choice.

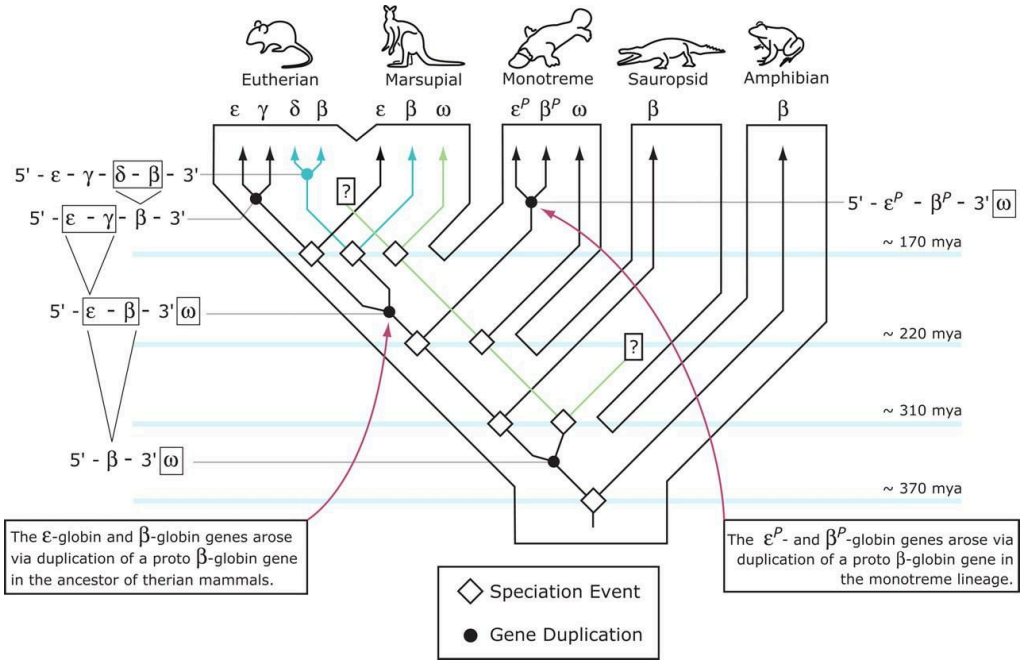


Figure 8: Homolog phylogeny of globin subunits (Opazo, Hoffman, and Storz 2008)

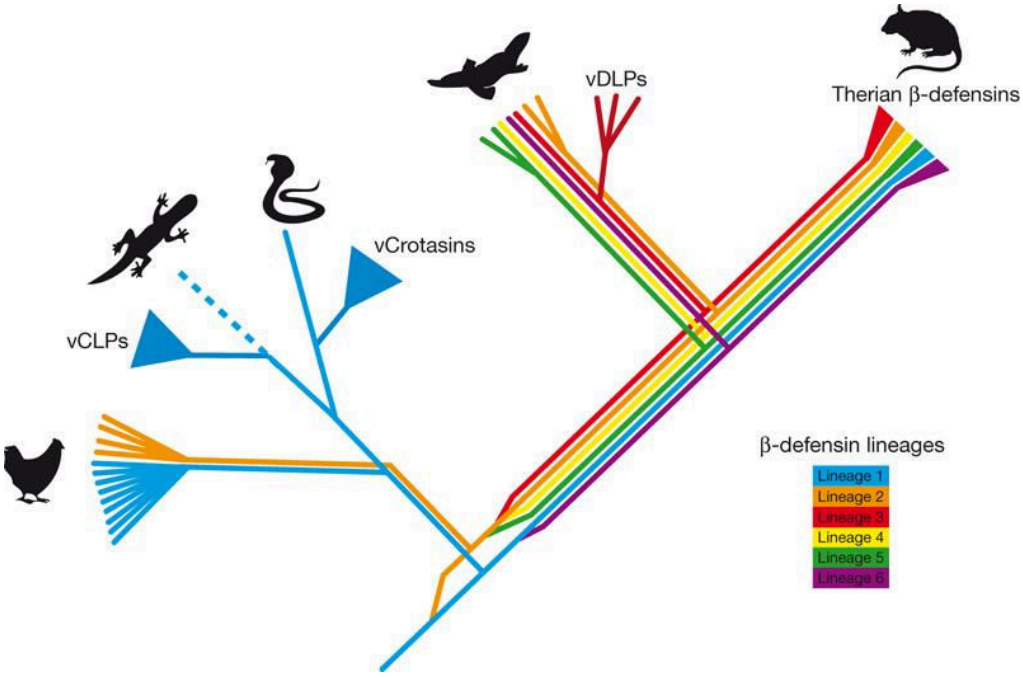


Figure 9: Homolog phylogeny of β-defensins (Warren 2008 *Nature*)

5 Collecting Sequence Data

5.1 Database searching

Often we will compile a dataset of homologous sequences by starting with a gene of interest and searching a database for homologous sequences.

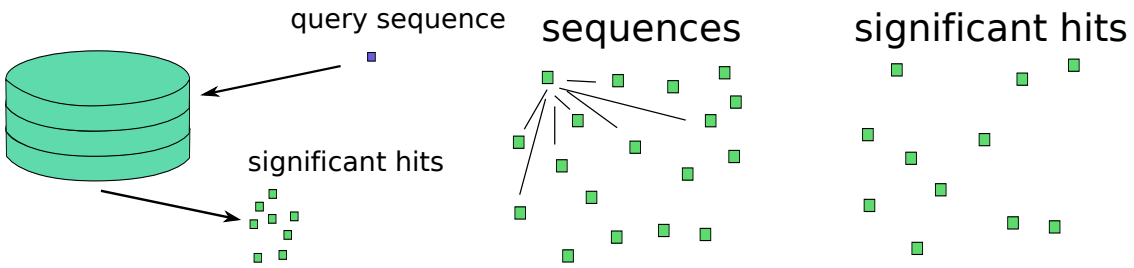


Figure 10: Diagrammatic representation of using a query sequence to search a database (*left*) or local set of sequence (*right*).

5.2 Where to get sequence data

- NCBI GenBank: https://www.ncbi.nlm.nih.gov/genbank/
- Ensembl: http://www.ensembl.org
- Many many other databases far too numerous to list here

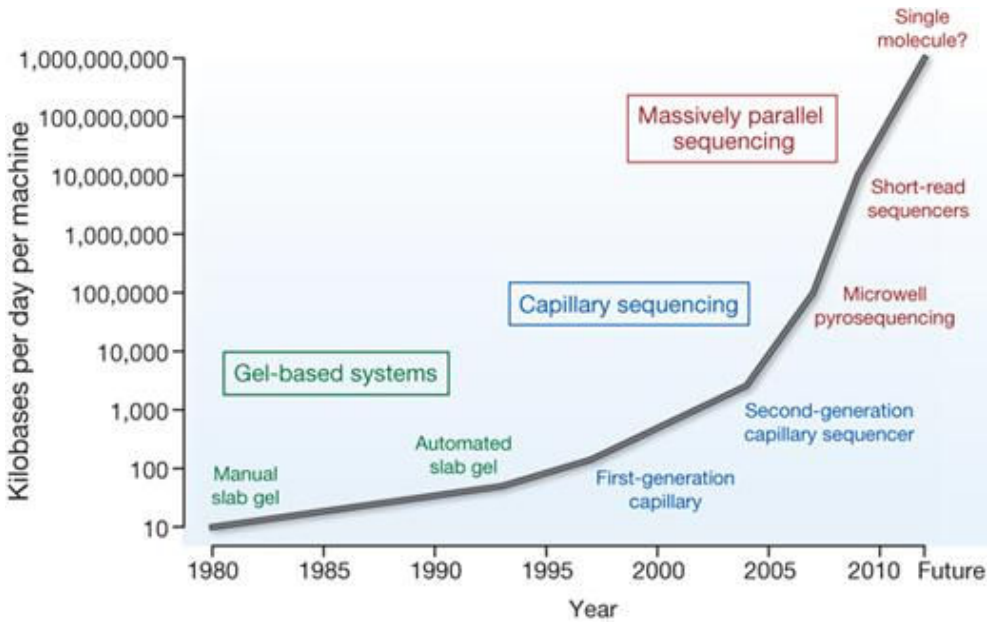


Figure 11: Increasing sequencing rate over time as different sequence technologies have developed.)

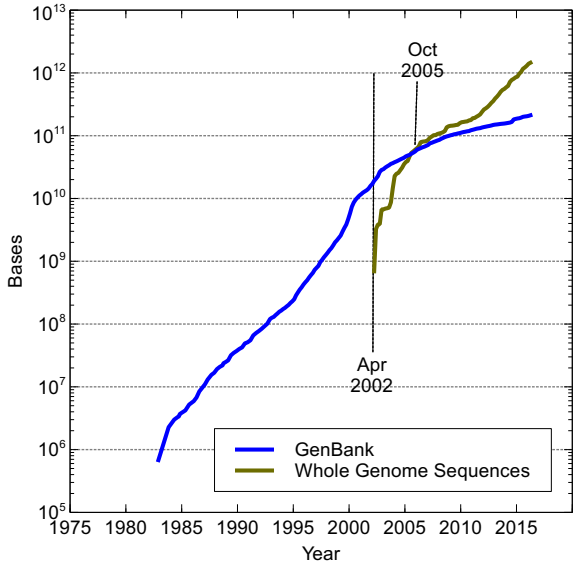


Figure 12: Amount of sequence data in GenBank increasing over time (Source: NCBI).

5.3 Can I download pre-made ortholog/paralog gene sequence sets?

Yes. . . but also no, or not necessarily comprehensively. Genes in GenBank are clearly indexed by species, but much more poorly indexed by gene homolog group. There are some curated sets of homologs or orthologs (will discuss later), but largely this will be up to you to do the work of determining what genes are homologous to your gene of interest. As will soon be apparent, **the process of determining homology is not straightforward and requires several subjective decisions.**

5.4 How many genes do I need to make a phylogeny or study evolutionary processes?

Depends on your question. Typical phylogenetic analyses will use 1–10 genes. Phylogenomic projects will use sometimes more than 20,000 genes.

# genes	Group	Study
1–10	Most taxonomic surveys	
17	Plants	Soltis et al. 2011
19	Birds	Hackett et al. 2007
150–30000+	Most genomic/trancsriptomic analyses	
140	Metazoa	Dunn et al. 2008
242	Metazoa	Ryan et al, 2011
1185	Molluscs	Smith et al 2011
2970	Seed plants	Lee et al. 2011
8251	Birds	Jarvis, et al. 2014
20374	Equids	Jónsson et al. 2014

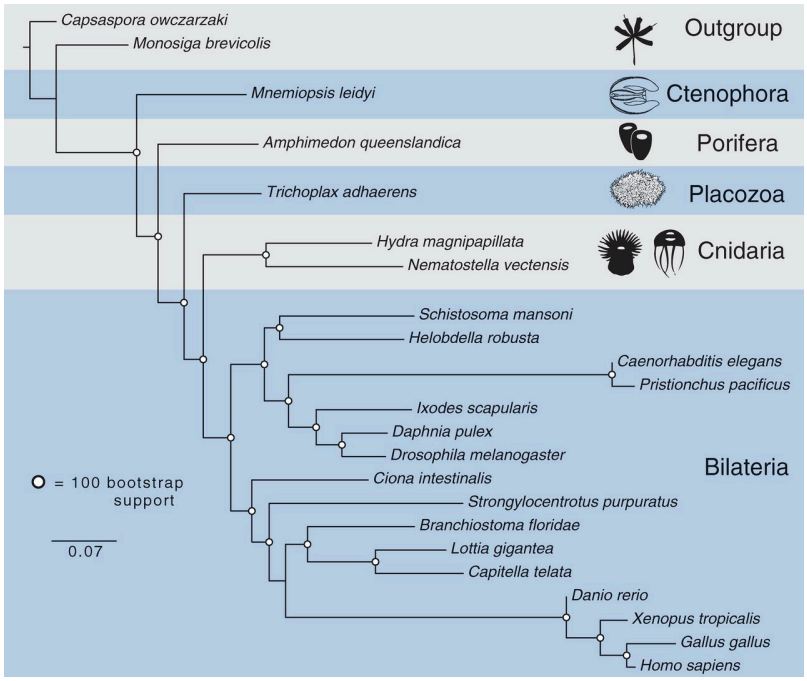


Figure 13: A phylogeny of metazoa using 242 genes (Ryan, et al. 2011).

5.5 Which genes/loci/regions should I use to study group _____?

It depends! The key is that you need to be able to sample homologous sequences from each individual you want to include in the same analysis.

5.5.1 ✎ How does the evolutionary distance of species impact the choice of sequences to use for a molecular evolutionary study?

5.5.2 ✎ What sequences would be appropriate study mice and rats? bacteria and humans? viruses and olives?

6 Homology Assessment and Sequence Alignment

6.1 Alignment as a tool for homology assessment

How do we determine whether sequences are homologous or not? The usual method involves attempting to align the characters in the sequence and see how well they align. All other things being equal, we assume **similar sequences should have stronger alignment than sequences that are not similar**.

6.2 Is homology a quantitative measure?

6.2.1 ✎ Can sequences be 90% homologous"?

Homology is boolean (true/false). **Sequences are either homologous or not**, and there are no "degrees" of homology. You might have different degree of support or degree of sequence similarity (a common supporting measure). However, this is one of many cases in molecular evolution where the true variable (homology) is inferred by an **indirect proxy measure** (similarity). **Similarity can have degrees, homology is true/false.**

Alignment Type	Algorithm	Search Type	Sequence Compared
Global Pairwise	Needleman-Wunch-Gotoh	Exhaustive	All characters
Local Pairwise	Smith-Waterman	Exhaustive	Most similar regions
BLAST	BLAST	Heuristic	Highly similar regions
Score-based Clustering	MCL, etc.	Optimization	Depends on scores used

A common means of homology detection is **pairwise alignment** between two sequences at a time. A **global alignment** is conducted if all characters in both sequences are aligned. However, **local alignment** of a highly-similar sub-sequence is the more commonly used alignment type, often in the form of a **heuristic local alignment** called **BLAST** that quickly finds highly similar regions.

If we want to compare many sequences to each other (many-to-many instead of one-to-one), we have to use a **clustering method** that simultaneously compares all sequences to form homologous sequence groups or "clusters." Often for clustering methods, a matrix of pairwise scores among all sequences is often used.

NOTE: Pairwise alignment methods will be the subject of the first tutorial. Clustering methods will be the subject of the second tutorial.

6.3 What we want to know...

From a biological perspective:

- Are these sequences homologous?
- Do they share common ancestry?

From an informatics perspective:

- Are these sequences "similar enough"? (proxy for homology)
- Should a sequence be removed? (poor information, error, etc.)

7 Pairwise Alignment

7.1 Basic pairwise alignment

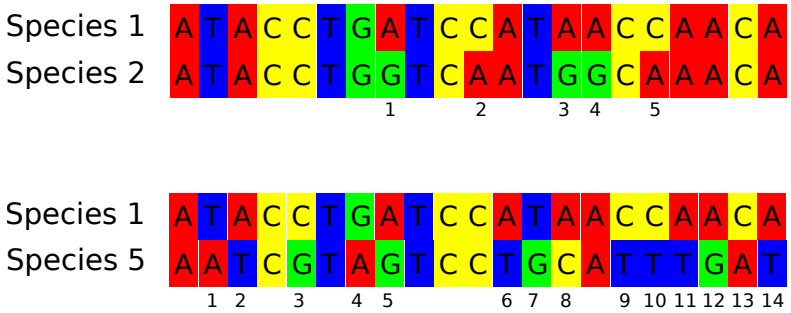


Figure 14: Pairwise comparison of two pairs of DNA sequences. Species 1 differs from Species 2 at 5 positions and from Species 5 at fourteen positions.

Consider these two sequences of letters that form common English words:

```
Species 1: SOMEONE
Species 2: AWESOME
```

When we align sequences, the goal is generally to **maximize the number of matching positions and minimize the number of non-matching positions** ("min-max"). So a simple alignment might look like this:

```
Species 1: ---SOMEONE
Species 2: AWESOME---
```

Now consider these three alignments of the same sequences:

```
Species 1: ACGTTAGA        -------ACGTTAGA        ACGTTAGA-
Species 2: CGTTGAA-        CGTTGAA--------        -CGTT-GAA
```

Technically, the left and middle *are* alignments. However, in neither case do we optimize the number of matching characters. The right side adds gaps carefully to maximize the number of matches and minimize the number of mismatches. **But how do we decide what makes one alignment "better" than another?**

7.1.1 ✎ If you wanted to give a quantitative "score" to an alignment what would be the positive and negative contributors to that score?

If we establish a simple scheme where matching position add $+1$, gaps are -1, and mismatches are not allowed.

```
Species 1: -------ACGTTAGA        ACGTTAGA-
Species 2: CGTTGAA--------        -CGTT-GAA

           Matches =   0          Matches = 6
           Gaps    =  15          Gaps    = 3
           Score   = -15          Score   = 3
```

7.2 ✎ Align the sequences below in a way that maximizes the score using the same $+1$ for matches and -1 for gaps:

```
Species 1: TTGGCACGTTAGA
Species 2: TGCACCTTAGTTA
```

7.3 How do we get the "best" alignment?

We cannot score all of possible alignments between two sequences, nor would that be efficient. We can find the best alignments using Dynamic Programming algorithms (sadly, we will not discuss these in detail here. There are great resources, if you are interested). Dynamic Programming solves a large problem by breaking it down and solving sub-problems.

One of the most famous dynamic programming methods is the **Needleman-Wunsch algorithm**. N-W is the standard global alignment algorithm, and has been cited over 11000 times since publication in 1970.

7.4 Gap models vary among models

7.4.1 Constant gap model

Each separate gap has a fixed penalty (regardless of size).

```
Species 1: GTTAGTTAC
Species 2: GTTA----C
match = 1, gap = −1, score = 4 (+5,-1)
```

7.4.2 Linear gap model (what we have done)

Each separate gap has a separate penalty, but proportionate to the size.

```
match = 1, gap = −1, score = 1 (+5,-4)
```

7.4.3 Affine gap model)

Each separate gap has one initial penalty for opening the gap AND a separate parameter for each extension of the gap.

```
match = 1, open = −2, ext = −1, score = -1 (+5,-2+(-1*4))
```

You can try your own at:
https://www.ebi.ac.uk/Tools/psa/emboss_needle/nucleotide.html

7.5 Protein alignment

Protein (amino acid) sequence alignment is not different from nucleotide in the algorithmic approach. The major difference is the scoring matrices. The standard choice for the scoring matrix is **BLOSUM** (BLOcks SUbstitution Matrix; "PAM" is another standard not often used anymore.) BLOSUM matrices are generated empirically from comparing actual homologous sequences, and calculating the log-odds of particular amino acid changes (e.g., R→K). The number after the BLOSUM refers to the percent identity of sequences used to generate the matrix (e.g., BLOSUM62 means only sequences with $\geq$62% matching characters were used).

	Ala	Arg	Asn	Asp	Cys	Gln	Glu	Gly	His	Ile	Leu	Lys	Met	Phe	Pro	Ser	Thr	Trp	Tyr	Val
Ala	4																			
Arg	-1	5																		
Asn	-2	0	6																	
Asp	-2	-2	1	6																
Cys	0	-3	-3	-3	9															
Gln	-1	1	0	0	-3	5														
Glu	-1	0	0	2	-4	2	5													
Gly	0	-2	0	-1	-3	-2	-2	6												
His	-2	0	1	-1	-3	0	0	-2	8											
Ile	-1	-3	-3	-3	-1	-3	-3	-4	-3	4										
Leu	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4									
Lys	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5								
Met	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5							
Phe	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6						
Pro	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7					
Ser	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4				
Thr	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5			
Trp	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11		
Tyr	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	
Val	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4

Figure 15: BLOSUM62 matrix of amino acid matching scores.

```
A-A = 4      Y-F = 3      P-F = -4      W-W = 11      C-C = 9
```

Primarily, these matrices end up accounting for the different side chain physiochemical properties that make some amino acids more similar to each other. You can see that an amino acid to itself is usually high scoring, as are amino acids with similar chemistry (F and Y). Rare residues with specific functional properties have the highest scores (C and W), since they change far more rarely.

As shown below, the choice of different BLOSUM matrices will change the best alignment, because you are altering the scoring system.

```
Species 1: HEAGAWGHEE        HEAGAWGHE-E          HEAGAWGHE-E
Species 2: PAWHEAE           --P-AW-HEAE          -P--AW-HEAE
           unaligned         BLOSUM62             BLOSUM30
```

8 Local Alignment

Smith-Waterman is the standard **local alignment algorithm** and was published in 1981 (cited over 10,000 times). This is also a dynamic programming algorithm. While global alignments must align the complete sequences, local alignment will return the best aligned subregion. A local region with maximum score under SW is called a **maximum-scoring segment pair (MSP)**. Thematically, this means that local alignment also includes **the length of aligned region itself as a "trade-offs."**

Here is an example of global and local alignment for nucleotides:

```
Species 1: ACGTTAGA       ACGTTAGA         GTTAGA
Species 2: GTTGAA         --GTTGAA         GTTAGA
           unaligned      global           local
```

And for amino acids:

```
Species 1: HEAGAWGHEE        HEAGAWGHE-E          AWGHE-E
Species 2: PAWHEAE           --P-AW-HEAE          AW-HEAE
           unaligned         global               local
```

9 Alignment Significance

9.1 How do I know when sequences are homologous?

It depends. When are sequences ***similar enough*** to be considered homologous? How do we quantify homology? Could sequences be similar by chance? How is the probability of this chance determined?

9.1.1 ✎ Are highly similar sequences necessarily homologous? Are homologus sequences necessarily similar? Construct a counter-example for both scenarios.

Take an example where we have two sequences A and B and we locally align them using Smith-Waterman and get a MSP alignment score of a value S. **Is the score S high enough to postulate with confidence that A and B are homologous?** This is unclear, because a similarity score cannot specifically tell us whether sequences are related by inheritance.

9.2 P-values and E-values

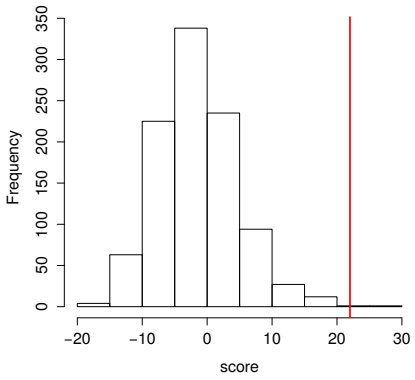
We cannot determine whether a score is absolutely strong or not. Therefore, we should ask in a more falsifiable way: **How many alignments do we expect by chance with at least the same score as our current alignment?**

9.3 Global Alignment Assessment

Needleman-Wunsch will **always give the best alignment (given the assumptions)**. In theory, no method allows usGlobal to analytically predict a distribution of possible alignment scores from random sequences. In practice, we can **generate a distribution simulated alignments** and get scores **compare our score to the distribution** if 100 random alignments give scores that are lower than the observed alignment score, then $P < 0.01$.

```
Species 1: HEAGAWGHE-E}
Species 2: -{}-P-AW-HEAE
BLOSUM62 and -2 gap penalty
Score = 22
```

The example to the right shows that only 1 out of 1000 scores was higher than 22, so our alignment is $P = 10^{-3}$.



9.4 Local Alignment Assessment

For a local alignment, we change our question to: **How many MSPs do we expect by chance with at least the same score as our current alignment?** Similar to the example above, we need to compare S to an expected distribution of scores to generate an E-value. **Given (1) an alignment score, (2) a query and database, and (3) a scoring system, the E-value represents the expected number alignments with a score greater than or equal to the given score.**

The E-value is typically calculated as:

$$E(S) = K(mn)e^{-\lambda S} \tag{1}$$

In Equation 1, m is length of query sequence, and n is the total length of database sequences. K is a scaling parameter based on properties of a database (i.e., modulates n), while λ is a scaling parameter based on the scoring system (i.e., modulates S). Both K and λ are analytically computed (usually implemented as a pre-computed table of values).

The probability of finding at least one alignment with our score (the P value) can be calculated from the E-values as:

$$P = 1 - e^{-E(S)} \tag{2}$$

So E and P decrease exponentially as score (S) increases (gets better), but E and P increase (get worse) as the database increases in size or the query sequence gets longer. This makes intuitive sense, a better score should be less likely but more local matches are possible in a large database with a long query sequence.

Score	E	P
39	12	0.99
41	2.9	0.95
42	1.4	0.76
46	0.08	0.08
49	0.01	0.01
55	0.0001	0.0001

$m = 980$
$n = 10,030,834,086$
$m \times n \approx 10^{13}$
$K = 1.37$
$\lambda = 0.711$

Note that **P-values are probabilities**, so must be on a $[0, 1]$ interval. **E-values are expected numbers of matches**, and so can take any value greater than zero. In practice, E-values of "zero" are reported when the value becomes so small that it experiences "floating-point underflow" by exceeding the computationally allowed number of decimal places (e.g, $< 10^{-250}$).

9.5 WARNING: E-values have no absolute meaning and are not comparable between different searches.

E-values are calculated based on the specific query sequence, database, and search parameters used, so they are **not directly comparable** between different searches. This means you could even run the same search a year from now on the GenBank database and it would potentially have a different E-value (if the database changed). However, **bit scores are normalized and thus comparable.**

Bit scores are calculated as:

$$S' = (\lambda S - \ln K)/\ln 2 \tag{3}$$

10 BLAST

10.1 BLAST is the fast search engine algorithm of sequence data

BLAST (Basic Local Alignment Search Tool; https://blast.ncbi.nlm.nih.gov/Blast.cgi) was published by Altschul et al in 1990 (the family of programs has been cited over 150,000 times). Smith-Waterman is often slow because it returns the *best* alignment (MSP) for any two sequences. BLAST *approximates* the best match (but does not guarantee it like S-W). BLAST calculates **"high-scoring segment pairs" (HSPs)** instead of the maximum segment pairs (HSP instead of MSP).

10.2 Basic BLAST Process

- Filter out repeats and low-complexity regions (S-W does not do this).
- Break the sequence into "words" (k-mers) of a standard length (28 for nucleotides is standard.
- For each "word," predict common mutants in a one-step-mutation "neighboorhood."
- Organize the words into a form best for searching
- Scan the other sequence for words that match.
- Extend these matches in the local neighborhood (these are going to be HSP or high scoring segment pairs)
- Extension stops when the score decreases past a certain point (X) when compared to the highest score

10.3 Break the sequence into words and generate mutational neighbors

The sequence is broken into a set of "words" (overlapping k-mers), then a set of sequences that are "one mutation away" from the words is determined.

```
GTTCACATCATCCTGC
GTTC                    CTTC,GTTC,GATC,...
 TTCA                   TTCT,TTGA,TTGT,...
  TCAC                  AGAC,CCAC,TCTG,...
   CACA                 CACG,CATA,TACA,...
    ACAT                ...
     CATC               ...
      ATCA              ...
```

10.4 Scan other sequence to find HSPs and extend from them

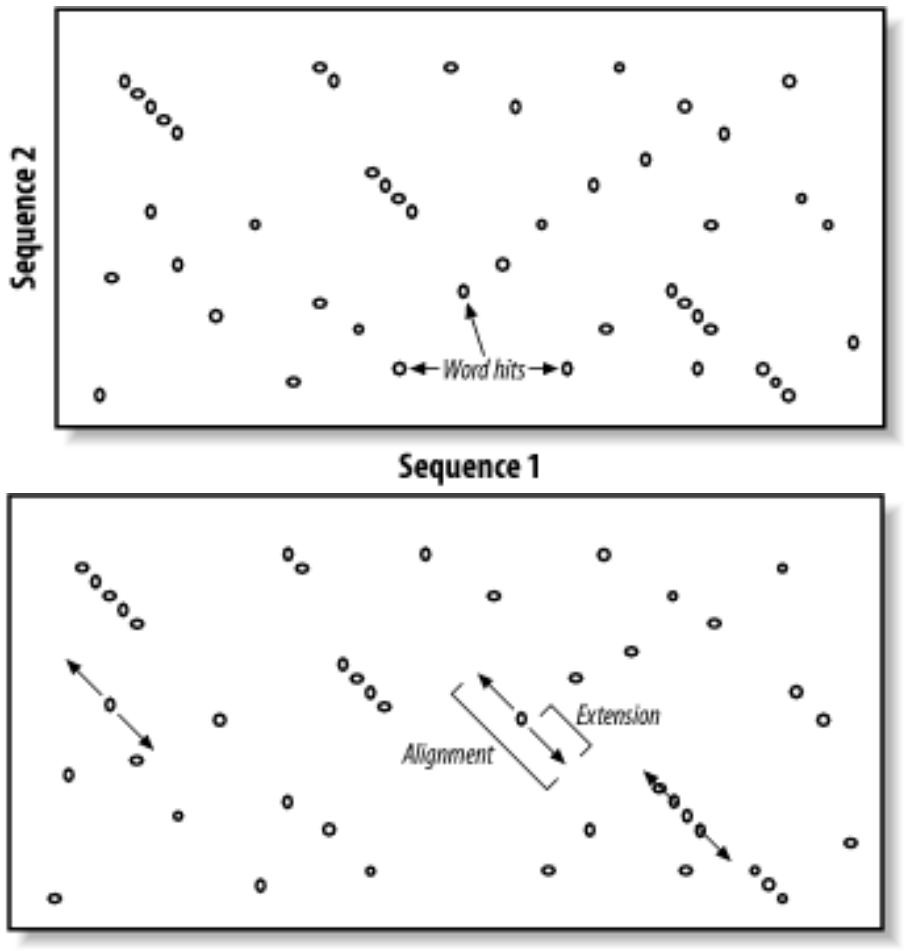


Figure 16: Representation of BLAST word comparison. In the diagram, the dots are "word" matches between the sequence. (Korf, et al.)

Below in Figure 17, the procedure for extension is shown. Once a word is found from the query sequence, the local alignment is "extended" out to include additional matches/mismatches/gaps until a threshold drop is reached (Figure 18). After the threshold score drop is reached, extension ends and the local alignment produced (or "hit") is scored by a **"bit score"**.

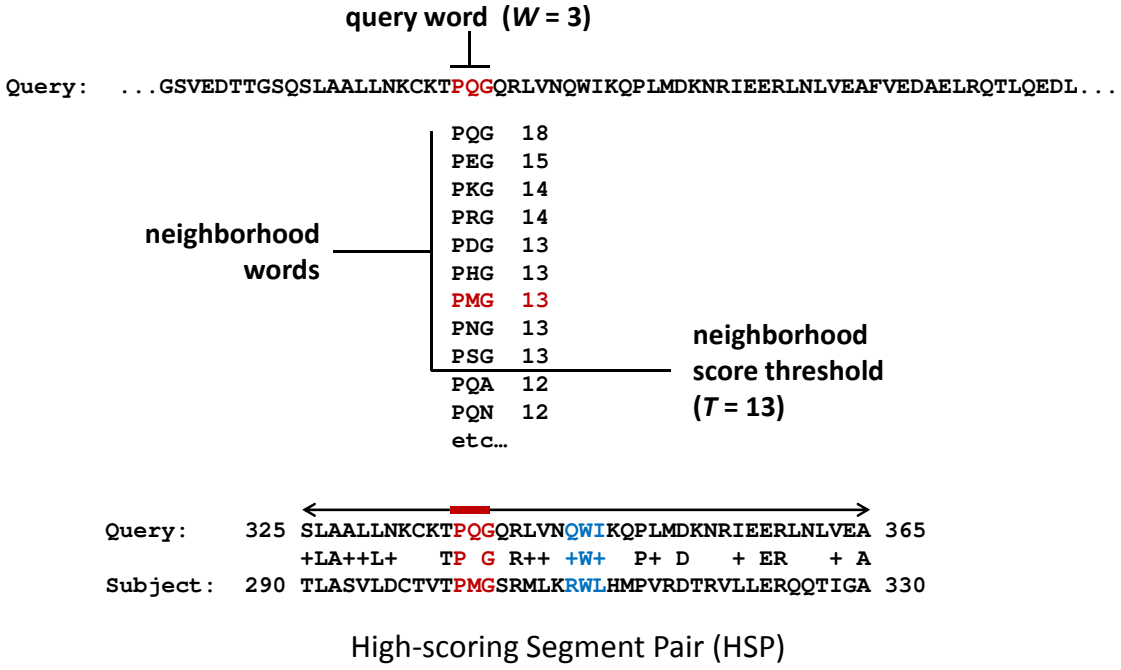


Figure 17: Representation of BLAST extension procedure. The seed word is shown in red (Altschul, et al.)

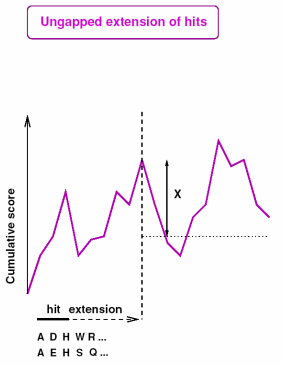


Figure 18: Diagram showing the change in score with additional extension of the alignment. After the sequence drops by a threshold value X, the heuristic search algorithm terminates.

10.5 How do we assess BLAST "hits"?

BLAST "hits" refers to all sequences that have HSPs with the query sequence. We can use the E-value (as with Smith-Waterman) to assess the significance of the BLAST bit score for a given sequence hit.

10.6 An example using 18S rRNA

Below is part of the sequence from GenBank for the 18S rRNA subunit from a specimen of *Abelia trifolia*. (https://www.ncbi.nlm.nih.gov/nuccore/AJ236004.1)

```
>AJ236004.1 Abelia triflora 18S rRNA gene
NNNNNNNNNNNNNNNNNNGTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTGTAAGTATGAAC
TAATTCAGACTGTGAAACTGCGAATGGCTCATTAAATCAGTTATAGTTTGTTTGATGGTACCTGCTACTC
GGATAACCGTAGTAATTCTAGAGCTAATACGTGCAACAAACCCCGACTTCTGGAAGGGATGCATTTATTA
GATAAAAGGTCGACGCGGGCTCTGCCCGTTGCTGCGATGATTCATGATAACTCGACGGATCGCACGGCCC
TCGTGCCGGCGACGCATCATTCAAATTTCTGCCCTATCAACTTTCGATGGTAGGATAGTGGCCTACTATG
GTGGTGACGGGTGACGGAGAATTAGGGTTCGATTCCGGAGAGGGAGCCTGAGAAACGGCTACCACATCCA
AGGAAGGCAGCAGGCGCGCAAATTACCCAATCCTGACACGGGGAGGTAGTGACAATAAATAACAATACCG
GGCTCTTTGAGTCTGGTAATTGGAATGAGTACAATCTAAATCCCTTAACGAGGATCCATTGGAGGGCAAG
TCTGGTGCCAGCAGCCGCGGTAATTCCAGCTCCAATAGCGTATATTTAAGTTGTTGCAGTTAAAAAG
```

10.6.1 ✎ Why would the BLAST alignment of the sequence with itself start at base position 20?

Here is the BLAST alignment of this sequence against another plant 18S sequence. Note that the bit score was 2802 and the E-value was 0.0 (i.e., smaller than 10^{-250}). The percent identity is 96%, meaning that 96% of positions had a match *within* the local alignment itself (NOT globally). Also note that the "strand" was Plus/Plus. If one sequence had a reverse complement match of the other this would be "Plus/Minus."

```
Score              Expect  Identities      Gaps        Strand
2808 bits(1520)    0.0     1670/1747(96%)  9/1747(0%)  Plus/Plus

Query  20    GTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTGTAAGTATGAACTAATTCAGA  79
             ||||||||||||||||||||||||||||||||||||||||||| ||||||||||||||||||
Sbjct  20    GTAGTCATATGCTTGTCTCAAAGATTAAGCCATGCATGTGCAAGTATGAACTAATTCAGA  79

Query  80    CTGTGAAACTGCGAATGGCTCATTAAATCAGTTATAGTTTGTTTGATGGTACCTGCTACT  139
             |||||||||||||||||||||||||||||||||||||||||||||||||||||||| ||||||||
Sbjct  80    TTGTGAAACTGCGAATGGCTCATTAAATCAGTTATAGTTTGTTTGATGGTATCTGCTACT  139

Query  140   CGGATAACCGTAGTAATTCTAGAGCTAATACGTGCAACAAACCCCGACTTCTGGAAGGGA  199
             |||||||||||||||||||||||||||||||||||| ||||||||||||||||||||||||
Sbjct  140   CGGATAACCGTAGTAATTCTAGAGCTAATACGTGCACCAAACCCCGACTTCTGGAAGGGA  199

Query  200   TGCATTTATTAGATAAAAGGTCGACGCGGGCTCTGCCCGTTGCTGCGATGATTCATGATA  259
             |||||||||||| |||||||| |  | || ||||||||| |||  | ||||||||||||
Sbjct  200   TGCATTTATTAGAAAAAAGGTC-AATCCGGTTCTGCCCGTCGCTCTGGTGATTCATGATA  258
```

```
 Score =  113 bits (61),  Expect = 1e-28
 Identities = 76/83 (92%), Gaps = 3/83 (4%)
 Strand=Plus/Plus

Query  1654  TCCTACCGATTGAATGGTCCGGTGAAGTGTTCGGATCGCGGCGACGTGGGCGGTTCGCTG  1713
             ||||||||||||||||||||||||||||||||||||||||||||||||  ||||||||  | |
Sbjct  1244  TCCTACCGATTGAATGGTCCGGTGAAGTGTTCGGATCGCGGCGACA-GGGCGGTTS-C-G  1300

Query  1714  CCGGCGACGTCGCGAGAAGTCCA  1736
             |||||||||| | ||||||||||
Sbjct  1301  CCGGCGACGTTGTGAGAAGTCCA  1323



Lambda      K        H
    1.33    0.621     1.12

Gapped
Lambda      K        H
    1.28    0.460    0.850

Effective search space used: 2298183



Matrix: blastn matrix 1 -2
Gap Penalties: Existence: 0, Extension: 2.5
```

Figure 19: Output from BLAST showing the λ and K parameters at the end.

query	subject	%ident	length	#mismat	#gp_open	que_sta	que_end	sub_sta	sub_end	evalue	score
18S_Abelia	18S_Abelia	100	1748	0	0	20	1767	20	1767	0	3229
18S_Abelia	18S_Acorus	96.41	474	16	1	76	548	1	474	0	785
18S_Abelia	18S_Acorus	92.72	508	36	1	896	1402	475	982	0	776
18S_Abelia	18S_Acorus	91.57	83	4	3	1654	1736	1244	1323	1.00E-28	113
18S_Abelia	18S_Aextoxicon	98.22	1741	31	0	24	1764	1	1741	0	3044
18S_Abelia	18S_Agave	97.3	1742	44	3	24	1763	1	1741	0	2955
18S_Abelia	18S_Ailanthus	98.09	1728	33	0	37	1764	1	1728	0	3009
18S_Abelia	18S_Alisma	95.91	1734	69	2	20	1751	10	1743	0	2808
18S_Abelia	18S_Alnus	97.15	1508	41	2	204	1710	3	1509	0	2545
18S_Abelia	18S_Amborella	96.04	1742	66	3	22	1761	1	1741	0	2832
18S_Abelia	18S_Angelica	98.51	1749	25	1	20	1767	17	1765	0	3085
18S_Abelia	18S_Anisophyllea	96.93	1106	33	1	24	1129	1	1105	0	1853
18S_Abelia	18S_Anisophyllea	97.7	609	13	1	1160	1767	1136	1744	0	1046
18S_Abelia	18S_Anisoptera	97.2	1747	46	3	23	1767	1	1746	0	2953
18S_Abelia	18S_Annona	95.23	1069	40	11	74	1141	39	1097	0	1681
18S_Abelia	18S_Aphanopetalum	97.71	1750	36	4	20	1767	2	1749	0	3009
18S_Abelia	18S_Arabidopsis	97.04	1046	29	2	723	1767	453	1497	0	1759
18S_Abelia	18S_Arabidopsis	96.4	444	13	3	20	462	20	461	0	728
18S_Abelia	18S_Aristolochia	93.32	449	16	5	91	539	1	435	0	652

Figure 20: Output table of BLAST hits sorted by decreasing percent identity.

10.6.2 ✎ Why does the fourth entry have a lower E-value?

10.7 Other BLAST databases

- NR: non-redundant amino acid sequences (default)
- many many model organisms
- BCT: bacterial sequences
- ENV: environmental sequences
- EST: expressed sequence tags
- GSS: genome survey sequences
- HTC: high throughput genomic sequencing
- TSA: transcriptome shotgun assemblies
- PAT: patent sequence
- VR: viral sequences

10.8 Many variations of BLAST searching exist

- **blastn**: nucleotide query to other nucleotide sequences
- **blastp**: protein query to other protein sequences
- **blastx**: nucleotide query is translated and used to search protein sequences database
- **tblastn**: proteins sequence is searched against translated nucleotides database
- **tblastx**: translated nucleotides searching against translated nucleotide database
- conserved domains
- vector screening
- MegaBLAST - essentially identical sequences BLAST searches

10.9 A brief demo

1. Open BLAST (https://blast.ncbi.nlm.nih.gov/Blast.cgi
2. Click "Nucleotide Blast" on the left
3. Copy/Paste the Sequence from *Abelia* earlier in this section into the box.
4. Select the radio button for "Somewhat similar sequences" toward the bottom.
5. Click the "BLAST" button.

At the top you should see a graphic with lots of red lines. These are visual representations of strong alignments ($\geq$ 200). Lower down, you should see a list of BLAST hits, which should all be 18S rRNA sequences from other plants. Note that the first entry is the sequence itself since we used a sequence from the database we searched. Click on the name of a sequence hit will show you the alignment. Clicking on the accession number at the right of each table entry will take you to the database entry for that sequence. Note the checkboxes on the left that allow bulk downloading via buttons at the top of the table.

10.10 Protein or DNA search?

Should you use blastn or blastp? There are 4 potential nucleotides $\{A, C, G, T\}$ and therefore four potential states. There are 22 amino acids states (including stops). Therefore, blastp should be more sensitive than blastn (larger state space leads to lower chance of a random hit). Overall, **if sequences are highly similar, DNA works well, otherwise use protein.** If no translated sequences available, then DNA is required.

11 Homology and Similarity (Revisited)

11.1 When sequence similarity imply sequence homology?

It depends. **Significant alignment** over the **majority of the length of both sequences** strongly *suggests* homology. However, ***homologous sequences do not always produce significant alignments.*** For example, regions with low complexity (but that are not cleaned out by initial steps in BLAST) can produce significant alignments with virtually no homology

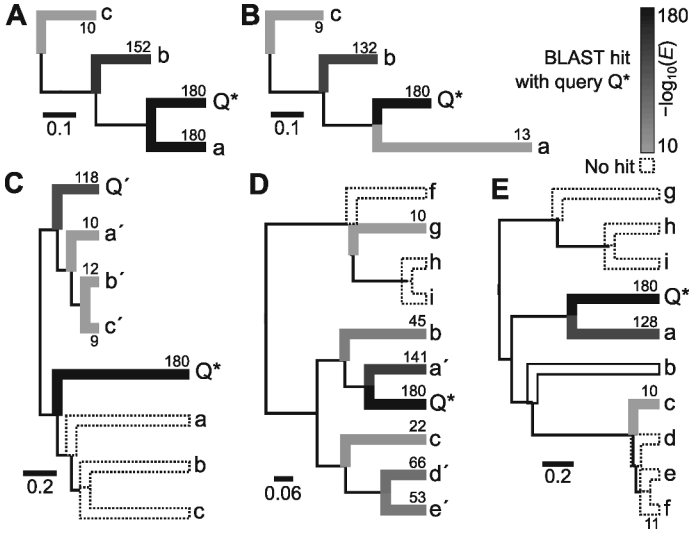


Figure 21: Some examples of how heterogeneous rates of sequence change can cause poor similarity between homologous sequences (Pease & Smith 2017. *Brief. Bioinform.*).

11.1.1 So what are the "rules" for determining homology?

There are no easy or standarized rules (decision difficult). That being said, the following might be considered generally conservative cutoffs:

Sequence type	Percent Identity	E-value
Nucleotides	>70%	$< 10^{-6}$
Proteins	>25%	$< 10^{-3}$

- **You must verify and thoroughly explore your own dataset.**
- **In a high-throughput large-scale analysis, there will be a (inescapable) margin of error.**

12 Conclusions and Questions

- In order to study **molecular evolution**, we need **homologous sequences** to model evolutionary processes that differentially affect sequences related to the **same ancestral sequence.**
- Homology is not directly knowable. We must rely on **similarity as a proxy**, and later phylogenetic relatedness and synapomorphy.
- Similarity is quantitatively measured by scoring **scored by sequence alignment**
- Similarity scores are **evaluated against a pool of other possible alignment scores** in a given database.
- Local alignment is the more common and efficient measure of similarity.
- Subjective cutoffs, parameters, and judgment calls are required to decide when "similar" becomes "homologous"

12.1 As you assemble your sequence sets from databases:

- Be critical and skeptical of sequences!
- Check for redundant database records (downloading the same sequence twice).
- Watch our for repetitive regions, paralogs, or fragmentary sequences.
- Recursively search for matches using your growing pool of sequences. (stepping stone searches)

12.2 Some other informatics principles we discussed:

- Trade-offs often occur when you are minimizing and maximizing multiple independent parameters.
- Often a proxy quantitative measure is used with cutoffs and thresholds to inform a true (but unknowable/incalculable value)
- Floating point underflow can occur when numerical values get very small.

Keep careful records as you go.

(Hint: At least an Excel table with sequence ids)

Automate, automate, automate.

Manual data entry leads to manual data error.

(Bulk downloading and copy/paste are your friends.)