ICQB

Introduction to Computational & Quantitative Biology (G4120)
Spring 2017

Oliver Jovanovic, Ph.D.
Columbia University
Department of Microbiology & Immunology

Lecture 5: Sequence Analysis
March 6, 2017

Sequence Analysis

Sequence analysis involves using a variety of techniques and algorithms to attempt to predict the function, structure or evolution of a DNA, RNA or amino acid sequence.

Examples

- Identifying intrinsic features such as open reading frames (ORFs), introns/exons, etc.
- Predicting molecular structure from the underlying sequence.
- Identifying single nucleotide polymorphisms (SNPs).
- Comparing a sequence to one or more other sequences to find similarities.
- Comparing a sequence to other sequences to construct a phylogenetic tree.

Intrinsic Sequence Analysis

Intrinsic sequence analysis involves evaluating sequence properties without explicitly referring to other sequences, using a derived formula, sequence or value.

Examples

With respect to DNA sequences, this includes calculating %G+C values, identifying an ORF, analyzing the information content of the sequence, or comparing the sequence against itself for the presence of direct or inverted repeats.

Extrinisic Sequence Analysis

Extrinsic sequence analysis involves evaluating sequence properties by explicitly comparing them to other sequences or sets of sequences.

Examples
With respect to DNA sequences, this includes sequence comparison, either pairwise dot matrix, pairwise optimal comparison (Needleman-Wunsch or Smith-Waterman), pairwise heuristic comparison (BLAST, BLAT, FASTA) or multiple sequence alignment to detect homology, insertions or deletions, as well as statistical or phylogenetic analysis.

Combined Sequence Analysis

A common mistake in sequence analysis is to use only intrinsic or extrinsic approaches to analyze sequence data. Combining intrinsic and extrinsic computational approaches can give results neither alone can approach.

Examples

Checking to see if the translation of an ORF matches a known protein, or making sure that a region with high coding potential (based on extrinsic data) has an underlying ORF.

Analysis of DNA Sequences

- GC content
- Information Content
- Dot Matrix Analysis (Self or Pairwise)
- Optimal Pairwise Alignment (Smith-Waterman, Needleman-Wunsch
- Heuristic Pairwise Alignment (BLAST, BLAT)
- Multiple Sequence Alignment
- Phylogeny

etc.

Information Content

Uncertainty
Uncertainty can be thought of as the number of yes/no questions required to identify the state something is in. It can be measured in bits.

- A coin toss, with only 2 possibilities, can be identified with a single question (i.e., "Is it heads?")
- A nucleotide, with 4 possibilities, can be identified with two questions (i.e. "Is it a purine? Is it adenine?")

Maximum Uncertainty

Maximum Entropy = $\log_2(n)$ where n is the number of possible states

Coin	$\log_2(2) = 1$ bit
DNA	$\log_2(4) = 2$ bits

Compression algorithms offer one approach to testing the randomicity of a DNA sequence. A very random DNA sequence will require close to 2 bits per nucleotide to represent it, even when compressed. A sequence of DNA that has repeating patterns, or is otherwise highly structured, should be capable of being represented by less than 2 bits per nucleotide.

Compression of DNA

Compression Results with Standard Compression Algorithms

RK2 (60,099 bp)	62,555 bytes	Uncompressed (8 bit ASCII)
Arithmetic coding (.bin)	15,025 bytes	2.00 bits per nucleotide
Stuffit Deluxe (.sit)	15,195 bytes	2.02 bits per nucleotide
WinZip (.zip)	14,915 bytes	1.98 bits per nucleotide
UNIX Compress (.z)	17,667 bytes	2.35 bits per nucleotide

Compression Results with Specialized Biological Algorithms

Biocompress 2 (Loewenstern & Yianilos, DCC97)	1.62 -1.92 bits per nucleotide
Expectation Maximization (Allison, Edgoose & Dix, ISMB98)	1.61 - 1.91 bits per nucleotide
Approximate Repeat Model (Stern, et. al & Dix, M&BP01)	0.61 - 1.59 bits per nucleotide

Standard compression algorithms are not particularly useful for such analysis, but specialized compression algorithms can be. The extent of the compression can yield information about how structured a sequence is, or identify the presence of even weak repeats. Eukaryotic sequences generally have more repeats than prokaryotic sequences, and are likely to demonstrate more compression.

Dot Matrix Analysis of DNA

Dot Matrix Analysis

A dot matrix analysis of a DNA sequence involves listing the sequence vertically and horizontally, either comparing it against itself or another DNA sequence, then placing a dot where a match occurs. With DNA sequences, identity results in a match. A stretch of matches appears as a long diagonal.

Filtering Results

A filter is used to filter out noise and focus on regions with extensive matches. The filter uses a sliding window of a given window size (**W**), placing a dot where a certain number of matches, called the Stringency (**S**), occur within that window.

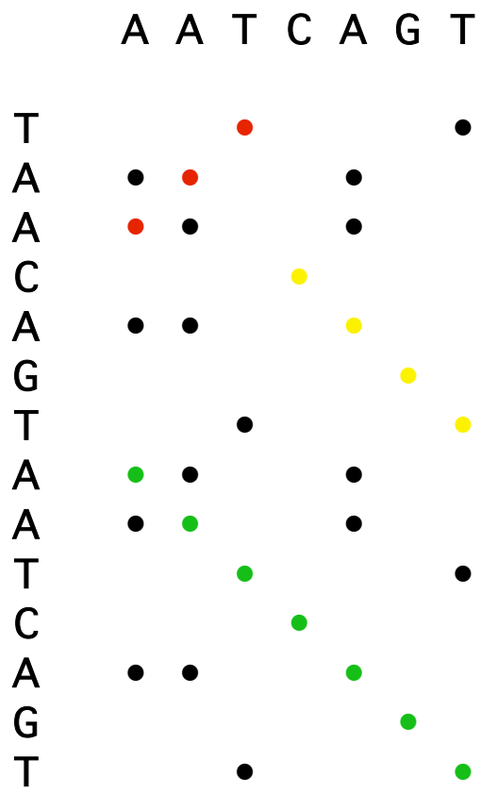
Direct and Inverted Repeats

Direct repeats appear as short parallel diagonals, while inverted repeats appear as short perpendicular diagonals.

Locating Repeats

If significant direct or inverted repeats are identified, they can be quickly located on the actual sequence using an appropriate sequence analysis program and function (i.e. the Seek Repeats and Seek Hairpins functions of DNA Strider).

Dot Matrix Analysis of DNA Illustrated

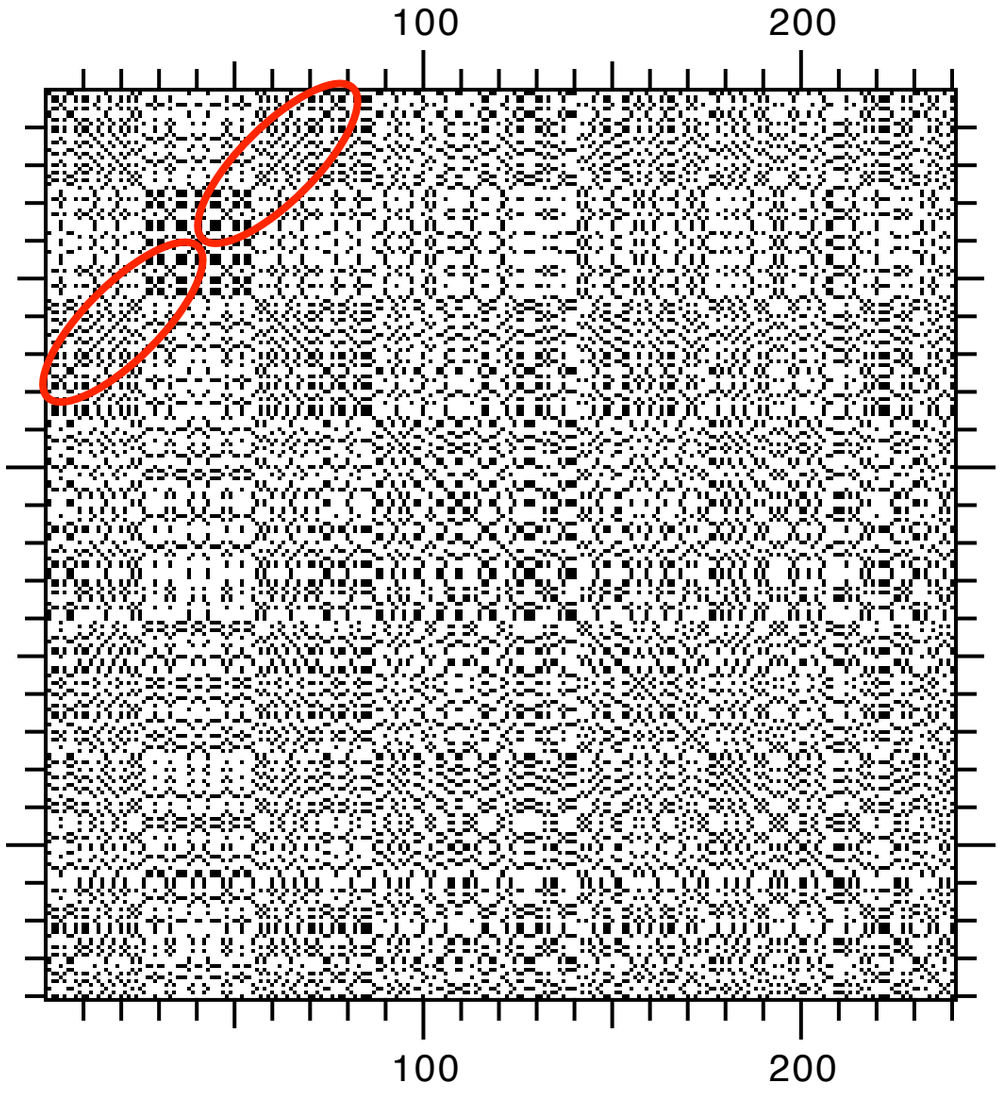


A dot matrix shows all possible matches between two sequences with a dot placed at every match.

- = Aligned sequence
- = Direct repeat
- = Inverted repeat

Dot Matrix Self Analysis of DNA

The 40 nt inverted repeat appears as short diagonals perpendicular to the diagonal of the sequence.



*This is a DNA Strider 1.3 DNA Self Matrix with Window Size 1 and Stringency 1 (**W1, S1**) of 200 nt of pBR322 sequence with a 40 nt inverted repeat added to the beginning of the sequence.*

Dot Matrix Pairwise DNA Analysis

Dot Matrix Pairwise Analysis

A dot matrix analysis of two sequences functions by listing one sequence vertically and the other sequence horizontally, then placing a dot where a match occurs. A region of sequence identity will be revealed by a diagonal line of dots.

Pairwise Analysis Window Size and Stringency Values

For pairwise DNA comparisons, a good starting point might be a window size of 15 with a Stringency of 7 (**W15, S7**). In general, long windows with medium or high stringencies should be used (**W15, S7**; **W23, S15**; **W15, S10-11**; **W11, S7** or **W7, S4**).

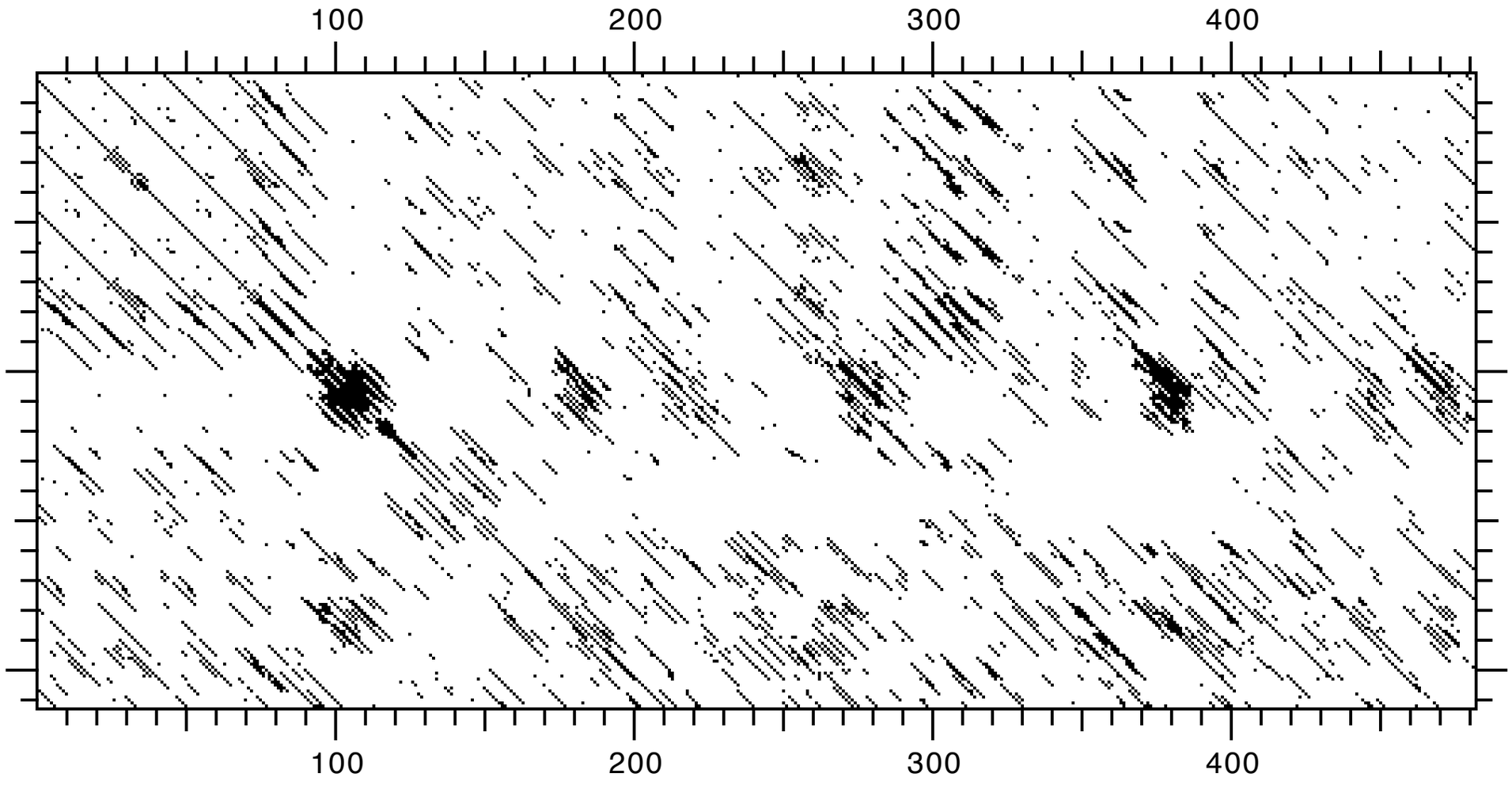
Orientation

Unless you are certain of the orientation, make sure to run one sequence antiparallel or run a DNA Antiparallel Matrix as well.

Dot Matrix Analysis and Alignment Algorithms

Since a dot matrix identifies all possible matches between two sequences, it can be helpful for making a first pass, then comparing the matches to the results of a sequence alignment algorithm such as BLAST, Smith-Waterman or Needleman-Wunsch to make sure all the matches are accounted for.

Dot Matrix Pairwise Analysis of DNA



*This is a DNA Strider 1.3 DNA Matrix with Window Size 15 and Stringency 7 (**W15, S7**) of two plasmid oriV regions (RSF1010 and R1162 (antiparallel)).*

Analysis of RNA Sequences

Transcription

- **Promoters** (e.g. consensus, sequence matrices, neural nets)
- **Transcriptional Terminators** (e.g. hairpins, TransTerm)
- **Splice Sites** (e.g. consensus, GeneSplicer)

Translation

- **Ribosome Binding Sites** (e.g. consensus, sequence matrices)
- **Coding Region Prediction** (e.g. ORFs, %GC, hexamer frequency, uneven positional base frequency, HMM)

Secondary Structure

- **Hairpins** (e.g. DNA Strider, GCG StemLoop)
- **Folding** (e.g. MFold)

etc,

Transcriptional Initiators

Prokaryotic Promoters

- Prokaryotic sigma 70 (σ^{70}) promoters are characterized by a **-35** consensus sequence (**TTGACA**) and a **-10** consensus sequence (**TATAAT**) which are respectively located 35 bp and 10 bp upstream of the transcriptional start point, which is labeled +1 (no 0 exists in transcriptional nomenclature).
- The consensus sequences vary for promoters using other sigma factors, and they also vary somewhat from species to species.
- They can be detected by consensus or matrix searching using software such as SeqMatrix or the use of specialized software trained to recognize prokaryotic promoters, such as a neural network.

Eukaryotic Promoters

- Three kinds of promoters exist for each major eukaryotic RNA polymerase (I, II and III):

I) **Pol I** promoters are associated with rRNA genes, and have a **GC rich Upstream Control Element** at -170 to -110, and a **core promoter element** at -40 to +20.

II) **Pol II** promoters have a **TATA box** at -25, a **CAA initiator sequence** at +1, upstream **GC rich elements**, an upstream **CCAAT box**, and **enhancer elements** that can be kilobases away from the +1.

III) **Pol III** promoters consist of at least three types, two with a **pair of control elements** at +50 to +100, and the last type with **three upstream control elements.**

- Although the procedures are more involved, since their structure can be complex and varies from one class to another, they can be detected by consensus or matrix searching or the use of specialized software trained to recognize a particular kind of eukaryotic promoter, such as PROSCAN and Pol3Scan.

E. coli σ^{70} Promoter

Consensus

```
 -35                    -10
TTGACA................TATAAT
```

Matrix

```
                                  -35 Region
                                 T  T  G  A  C  A
A   11 8  8  7  8  7  3  5  5  0  1  0  14 5  9  5
C   3  4  2  4  4  3  5  2  8  1  1  2  3  11 2  5
G   3  2  4  2  4  5  5  5  5  2  1  17 1  2  3  3
T   4  7  7  8  5  6  8  9  3  17 18 2  4  3  7  9

              Spacer Region
Length   9  10  11  12  13  14  15
         1   6  14   6   1   1   1

                     -10 Region
                    T  A  T  A  A  T
A   4  5  3  4  4  0  20 5 12 11  0  7  4  6
C   5  4  5  4  5  2  0  3  3  4  1  2  7  6
G   2  5  5  8  7  2  0  3  3  3  0  6  5  6
T   10 6  8  5  6  17 1  9  3  4  20 6  5  4
```

Transcriptional Terminators

Rho Independent Prokaryotic Terminators

- These are characterized by a hairpin loop structure followed by a string of approximately six **U**s in the RNA transcript. The hairpin loop is typically centered about 20 to 30 bases upstream of the last nucleotide in the transcript.
- They can be detected by searching for hairpin loops (using DNA Strider's Seek Hairpin function, or the GCG StemLoop program), then looking for a nearby string of **U**s, or by using specialized software such as TIGR TransTerm.

Rho Dependent Prokaryotic Terminators

- These require the trans-acting Rho protein factor, and often have a hairpin loop structure similar to that of a *rho* independent terminator, but no string of **U**s, and are characterized by a *rut* (rho utilization) consensus sequence of roughly 85 nucleotides starting approximately 100 bases upstream of where transcription terminates.
- The *rut* site can be detected by consensus or matrix searching.

Eukaryotic Terminators

- A 20 to 200 nucleotide poly-A tail is added to the 3' end of mRNA approximately 20 bases downstream of a **AAUAAA polyadenylation signal** consensus sequence.
- The signal can be detected by consensus or matrix searching.

RNA Splice Sites

- Splice sites typically consist of an upstream **AGGU donor site** consensus sequence and a downstream **AGG acceptor site** consensus sequence, from which an intron is spliced out to begin with **GU..** and end in **..AG**. In addition, a **branch site** consensus sequence **UAUAAC** is located 20 to 50 nucleotides upstream of the acceptor site.
- A fair amount of variation exists. Some donor sites end in **..GC**, and the exact residues conserved in a site can vary from species to species.
- At least one alternative splicing pathway exists which prefers different donor and acceptor sites (spliced to begin with **AU..** and end in **..AC**).
- Splice sites can be detected through consensus searching, matrix searching with software such as SeqMatrix, or the use of specialized software trained on a particular species such as GeneSplicer.

Translational Initiation

Prokaryotic Translation Initiation

- In prokaryotes, the **AGGAGG** Shine-Dalgarno consensus sequence is located 4 to 7 nucleotides 5' of the translation initiator **AUG** of most mRNAs.
- The sequence is complementary to a **CCUCCU** sequence at the 3' end of 16S rRNA.
- Other residues in the translation initiation region are also conserved, varying in detail from species to species.
- Prokaryotic translation initiators can be detected by consensus or matrix searching for the Shine-Dalgarno sequence, start codon, and other conserved residues using software such as SeqMatrix, or by the use of specialized software trained on prokaryotic translation initiation sequences such as the W101 Preceptron neural network algorithm.

Eukaryotic Translation Initiation

- In eukaryotes, translation typically initiates at the 5' **AUG** in the mRNA, and is not obviously complementary to rRNA, but also features other conserved residues.
- Initiators can be detected by consensus or matrix searching for the start codon and other conserved residues, or the use of specialized gene finding software trained on eukaryotic sequences.

Prokaryotic Ribsome Binding Sites

Shine-Dalgarno Consensus

AGGAGG....AUG

Plasmid RK2 Ribosome Binding Site Matrix

	A	G	G	A	G	G							C	A	A	U	G	A	A		
A	5	8	6	4	8	8	3	7	7	6	6	7	8	2	7	21	0	0	14	10	7	8
C	6	5	3	3	4	1	2	3	3	5	2	7	7	12	5	0	0	0	3	8	5	4
G	6	5	10	12	9	12	15	7	6	4	7	4	4	5	4	0	0	21	3	3	3	9
U	4	3	2	2	0	0	1	4	5	6	6	3	2	2	5	0	21	0	1	0	6	0

Prokaryotic Translation Initiation Regions

E. coli Translation Initiation Regions

	A	A	U	U	A	U	G	G	C	U	A
A	6	8	5	5	14	0	0	6	5	6	7
C	3	3	3	3	0	0	0	1	6	2	4
G	2	2	2	2	1	0	15	6	2	1	3
U	4	2	5	5	0	15	0	2	2	6	1

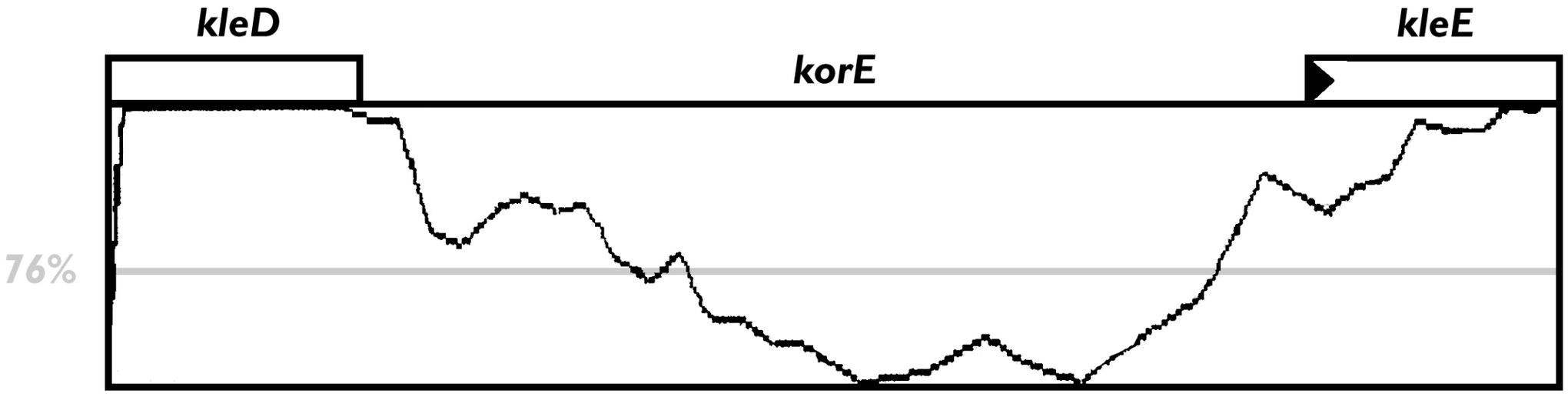
Plasmid RK2 Translation Initiation Regions

	C	A	C	A	A	U	G	A	A	A	G
A	7	8	2	7	21	0	0	14	10	7	8
C	7	7	12	5	0	0	0	3	8	5	4
G	4	4	5	4	0	0	21	3	3	3	9
U	3	2	2	5	0	21	0	1	0	6	0

Coding Region Prediction

- Due to the restraints imposed on them by having to code for the triplets of the genetic code, coding regions differ from non-coding regions in their nucleotide distributions.
- As a result, certain forms of computational analysis, including %G+C content, codon usage, hexamer frequency analysis (70-80%) neural nets (81-96%), hidden Markov models (92-99%) and uneven positional base frequency can theoretically distinguish coding regions from non-coding regions.
- Coding sequences typically have a nonrandom distribution of bases, with an uneven distributions of bases in each codon position.
- This allows for more sophisticated prediction of coding regions and mRNAs than simply looking for an ORF, which may or may not actually be a coding region.
- It can also be useful when trying to identify a regulatory RNA or RNA with some function other than an mRNA.
- Various free software packages allow one to perform such analysis, including the Staden package, EMBOSS, and web based programs such as FramePlot, GeneHacker, GeneMark, GENSCAN and GRAIL.

Uneven Positional Base Frequency Analysis



The method used here is that of Staden, and goes through the sequence in a single frame, counting the number of times each base appears in every one of the three possible codon positions and then compares these relative abundances. The more the relative abundances differ from each other (i.e. the less random the distribution), the more likely it is for the region to code for a protein. The method cannot predict the reading frame or strand. The gray line across the window represents a level that is exceeded by 76% of known coding sequences. In general, the higher the score (represented by the black line), the more likely a region is to code for a protein.

The region analyzed here is a sequence of 300 nt which as marked, contains the korE *region, and to either side portions of the* kleD *and* kleE *genes, marked as boxes, with filled triangles indicating the direction of translation. Both* kleD *and* kleE *are known coding regions for which a polypeptide product has been observed.*

GeneMark

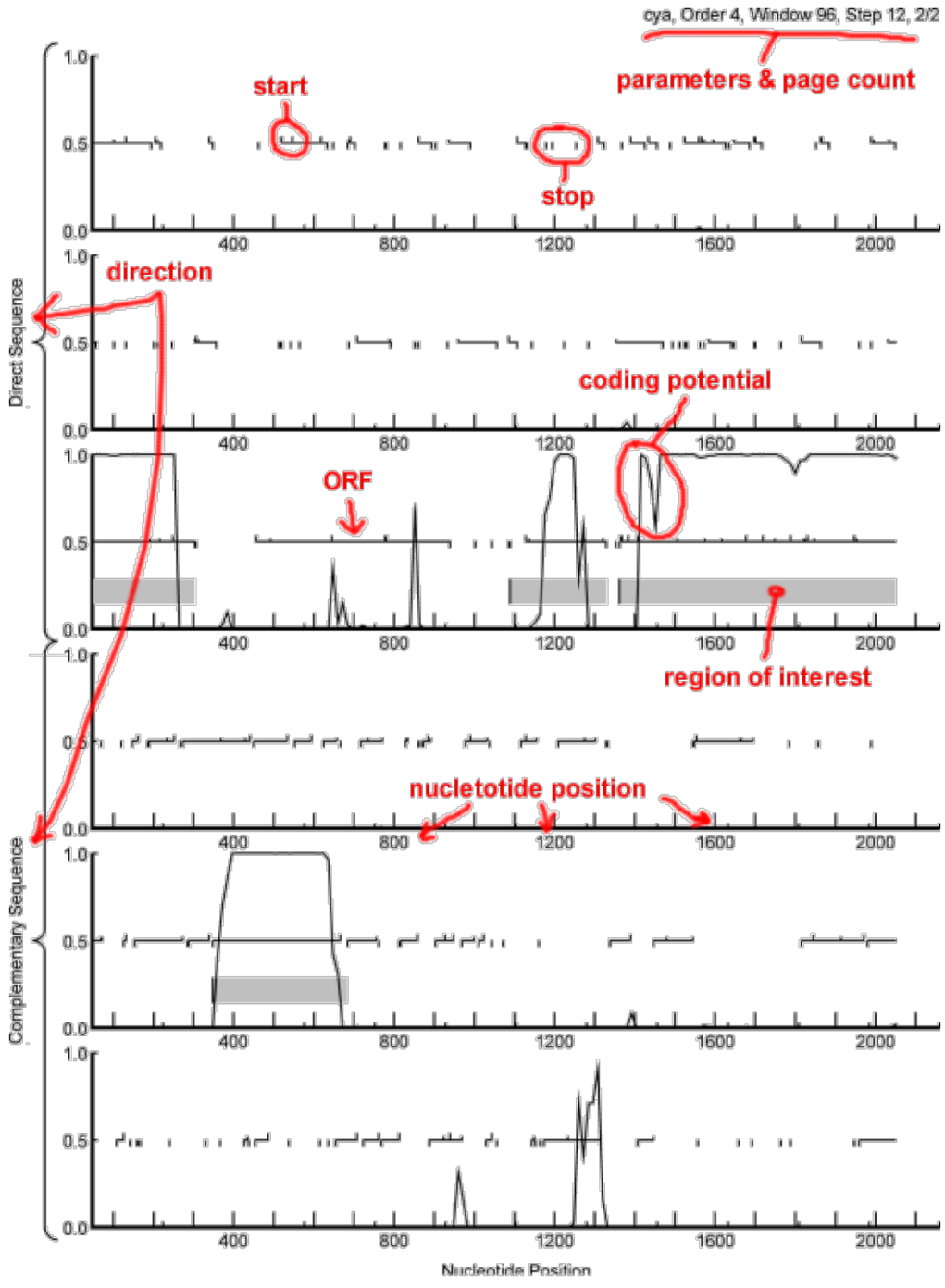
GeneMark and GeneMark.hmm
Mark Borodovsky, Georgia Institute of Technology
http://opal.biology.gatech.edu/GeneMark/

GeneMark
GeneMark evaluates the protein-coding potential of a DNA sequence (within a sliding window) by using Markov models of coding and non-coding regions for various species. This approach is sensitive to local variations of coding potential, and the GeneMark graph shows details of the coding potential distribution along a sequence.

GeneMark.hmm
GeneMark.hmm predicts genes and intergenic regions in a sequence as a whole using hidden Markov models with a hidden state network reflecting the "grammar" of gene organization. It identifies the most likely parse of the whole sequence into protein coding genes (with possible introns) and intergenic regions.

The two approaches can be run in parallel, using a combination of GeneMark-P and GeneMark.hmm-P for prokaryotic gene prediction, or GeneMark-E and GeneMark.hmm-E for eukaryotic gene prediction.

GeneMark Example



RNA Secondary Structure Prediction

- Individual hairpin loops are simple to predict (e.g. using DNA Strider Seek Hairpins) but may not accurately reflect the structure of an entire RNA.
- Accuracy of prediction is limited by accuracy of free energy calculations. Different tables of free energy values give different results.
- *In vitro* folding may be affected by the direction of synthesis, interaction of unpaired loops, interaction with other RNAs, interaction with other proteins, or environmental conditions.
- Modification of bases, such as which occurs with tRNAs, may affect folding.
- There may be multiple structures with similar or equally favorable free energy states.
- The structure prediction with the lowest free energy is not necessarily the functional conformation of the RNA.

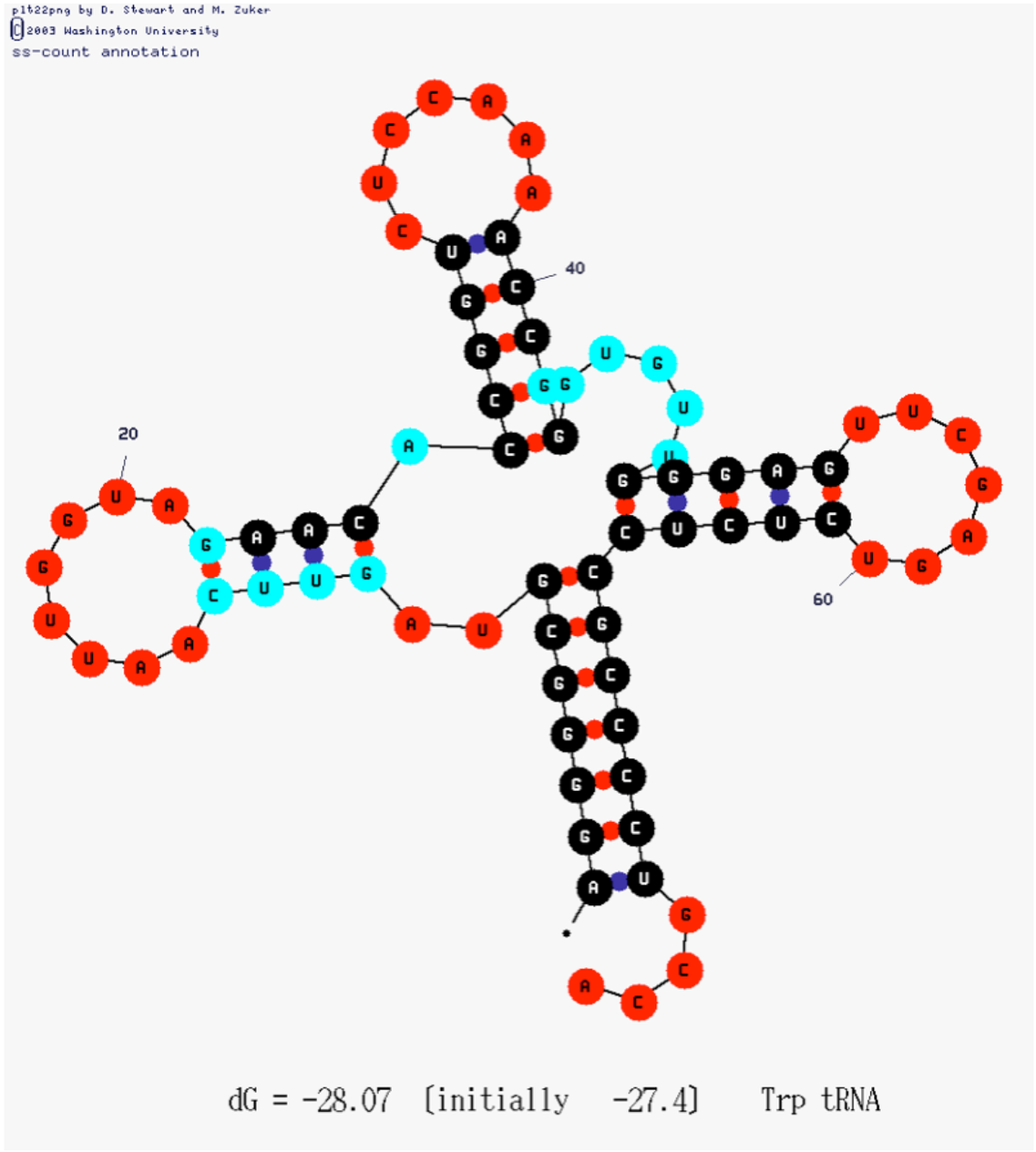
mfold

mfold (Michael Zucker, RPI)

- Predicts optimal and suboptimal RNA secondary structures based on a table of free energy, looking for the structure with the lowest total free energy (ΔG) value.
- Tries to find the base pairings that vary the least in a series of optimal and suboptimal RNA secondary structures.
- Can also be used to find DNA secondary structures.
- Can run mfold and display output with plotfold as UNIX applications.
- Can run mfold on the web at:

http://unafold.rna.albany.edu/?q=mfold

mfold Tryptophan Transfer RNA 2° Structure Prediction



Analysis of Protein Sequences

Coding Region Prediction
Start/Stop, Uneven Positional Base Frequency, Hexamer Frequency, Hidden Markov Models (HMM)

Protein-Protein Comparison
Dotplots, Needleman-Wunsch, Smith-Waterman, FastA, BLAST

Functional Region Prediction
Motifs, Profiles, Hidden Markov Models

Secondary Structure Prediction
Kyte-Doolittle, Chou-Fasman, Markov Models, Neural Nets

Tertiary Structure Prediction
Threading, Homology Modeling, Model Verification, Ab-initio Modeling

Multiple Sequence Alignment and Phylogeny
MSA, Clustal, Phylogenetic Trees

Protein-Protein Comparison

Dot Matrix
Can compare proteins to each other using a dot matrix. When doing so, use a small window size (W1-3) and low stringency (S1-2).

Needleman-Wunsch (global)
Performs an optimal global alignment of two protein sequences.

Smith-Waterman (local)
Performs an optimal local alignment of two protein sequences, useful for comparing conserved domains.

FastA (heuristic)
An older fast heuristic algorithm for comparing proteins.

BLAST (heuristic)
The most popular fast heuristic algorithm for protein comparison.

BLAT (heuristic)
A very fast heuristic algorithm related to BLAST and easy to run locally.

Protein Functional Region Prediction

Motif
Uses a pattern derived from a number of known examples of a functional protein region. As this yields only a single consensus sequence, it is less accurate than the Profile or HMM methods.
Example: PROSITE (**http://prosite.expasy.org**)

Profiles
Profiles are **statistical matrices** based on a family of known functional protein regions. They are more accurate than searching with a single consensus sequence.
Examples: Pfam (**http://pfam.xfam.org**)
CDD (**https://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml**)

Hidden Markov Models (HMM)
Hidden Markov models can be used to create statistical descriptions of a functional protein sequence family's consensus, which can then be used to accurately search for related functional domains.
Examples: SMART (**http://smart.embl-heidelberg.de/**)
HMMER (http://hmmer.org)

Other Resources
The NCBI (**http://www.ncbi.nlm.gov**) and the Center for Biological Sequence Analysis (**http://www.cbs.dtu.dk/services/**) maintain resources for identifying protein sequence features. In addition, ExPASy maintains an extensive list of protein resources curated by Amos Bairoch (http://www.expasy.org/links.html).

Protein Domain Databases

CDD (Conserved Domain Database)
CDD is an NCBI database that contains conserved domains based on recurring sequence patterns or motifs derived from two popular collections, Smart and Pfam, as well as contributions from NCBI, such as COG. The source databases also provide descriptions and links to citations. Since conserved domains correspond to compact structural units, CDs contain links to 3D-structure via Cn3D whenever possible. Conserved Domains are indexed for retrieval by keywords; links between Conserved Domains and Proteins, PubMed, and Taxonomy have been added. Conserved Domains are also linked to other Conserved Domains by two different neighboring mechanisms. "Similar" domains are defined as those giving overlapping annotations on sets of protein sequences, "Co-occuring" domains are defined as those giving non-overlapping annotations on sets of protein sequences.
https://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml

CD-Search (Conserved Domain Search)
CD-Search identifies conserved domains in a protein sequence by employing the reverse position-specific BLAST algorithm. The query sequence is compared to a position-specific score matrix prepared from the underlying conserved domain alignment. Hits may be displayed as a pairwise alignment of the query sequence with a representative domain sequence, or as a multiple alignment. CD-Search now is run by default in parallel with protein BLAST searches.
https://www.ncbi.nlm.nih.gov/Structure/cdd/wrpsb.cgi

CDART (Conserved Domain Architecture Retrieval Tool)
CDART allows one to search for proteins with similar domain architectures. It uses precomputed CD-Search results to quickly identify proteins with a set of domains similar to that of the query.
https://www.ncbi.nlm.nih.gov/Structure/lexington/lexington.cgi?cmd=rps

3D Protein Structure Databases

MMDB (Molecular Modeling DataBase)
MMDB is the NCBI protein stucture database. It consists a subset of experimentally determined three-dimensional structures obtained from the Protein Data Bank (PDB) which have had errors and ambiguities removed, and then were converted to ASN.1 (Abstract Syntax Notation 1) format. The data is available thru Entrez or the free Cn3D 3D structure viewer. MMDB currently contains over 10,000 structure records, with approximately 80% of the structures determined by X-ray diffraction studies, the rest by NMR or other experimental methods. Links are provided to Medline records and the NCBI taxonomy databases through PDBeast. Related sequences are provided by BLAST, related structures are provided by VAST.
https://www.ncbi.nlm.nih.gov/Structure/MMDB/mmdb.shtml

VAST (Vector Alignment Search Tool)
The structural data of proteins in MMDB are compared against each other using the VAST algorithm for detecting significantly similar substructures. Entrez or Cn3D can be used to retrieve structures which seem highly similar to the query protein structure, in much the same way as sequence neighbors computed by BLAST. This will retrieve almost all structures with an identical 3D "fold", even in distantly related proteins, though it may occasionally miss a few or report chance similarities.

VAST functions by reducing x, y, z coordinate data for all alpha helices and beta sheets in a protein into vectors, then creating pairs of vectors called secondary structure elements (SSEs), which it attempts to superimpose. It is a heuristic approach, not an optimal one, and loses some information by converting substructures to vectors, but is extremely fast.
https://www.ncbi.nlm.nih.gov/Structure/VAST/vastsearch.html

Protein 2° Structure Prediction

Kyte-Doolittle
A hydropathy plot that can indicate potential transmembrane or surface regions in proteins.
Accuracy: Scores -4.5 hydrophilic to 4.5 hydrophobic, poor for beta sheets (**DNA Strider**)

Chou-Fasman
A statistical approach to secondary structure prediction based on observed frequencies.
Accuracy: ~60% (**http://fasta.bioch.virginia.edu/fasta_www2/fasta_www.cgi?rm=misc1**)

Predict Protein
A secondary structure prediction method based on a consensus of several complementary prediction methods, including PHD, which uses jury decision between a number of neural networks, enhanced by multiple sequence alignment information.
Accuracy: ~80% (**http://www.predictprotein.org/**)

Protein 3° Structure Prediction

Homology Modeling
Builds a model of a protein based on homologies to proteins of known structure. Can produce good results when proteins with significant homology and known structure exist.
Examples: Modeller (**https://salilab.org/modeller/**)
SWISS-MODEL (**http://swissmodel.expasy.org/**)

Threading
Compares the fitness of protein sequence to assume various known tertiary structures. It assumes a particular fold, then evaluates the quality of the resulting structure. Can identify distantly related structural homologs and verify homology models.
Examples: 3D-PSSM, 123D, PHD (**http://www.predictprotein.org/**)

Model Verification
Checks the fitness of a protein sequence to assume a modeled fold.
Examples: VERIFY-3D, PROCHECK (**http://www.ebi.ac.uk/thornton-srv/software/PROCHECK/**)

Ab-initio Structure Modeling
Predicts a model of a protein directly from the sequence. Accuracy keeps improving.
Examples: RAMP, ROBETTA (**http://robetta.bakerlab.org/**)

Multiple Sequence Alignment

Multiple Sequence Alignment (MSA)
A multiple sequence alignment is an alignment of a set of sequences with structurally similar and evolutionarily homologous residues aligned in columns.

In an ideal alignment, columns of aligned amino acid residues would have similar locations in the 3D structure of a protein and would diverge from a common ancestral residue.

In theory, an unambiguously correct evolutionary alignment exists, but can be difficult to infer and computationally intensive to calculate. Where structural data is lacking or limited, as is generally the case, it is not possible to unambiguously identify structurally similar positions. Thus, defining a single unambiguous "ideal" alignment can be very difficult.

Identity: The extent to which two sequences are invariant.

Similarity: The extent to which sequences are related, based on sequence identity and/or conservation. Homology is often inferred from similarity, but homology only exists with common ancestry (i.e. *not* similarity from convergent evolution).

Conservation: Changes in an amino acid sequence that preserve the biochemical properties of the original residue. This is measured in most sequence comparison algorithms by substitution matrices in which scores for each position are derived from observations of the frequencies of substitutions in blocks of local alignments in related proteins.

Multiple Sequence Alignment Algorithms

Dynamic Programming vs. Heuristic Alignment

Using dynamic programming algorithms (such as Smith-Waterman or Needleman-Wunsch) to perform an optimal alignment of more than a few sequences is computationally intensive, and generally impractical for large sets of sequences or lengthy sequences.

As a result, most commonly used multiple sequence alignment algorithms take a heuristic approach. One common heuristic approach is progressive alignment, in which the problem is broken down into a series of pairwise alignments. The details of how to choose the initial pair to align, how to score alignments, how to align subsequent sequences, and whether subfamilies of alignments should be created can all vary.

msa (Dynamic)

This algorithm uses a technique that reduces the complexity of dynamic programming when applied to multiple sequences, and can give an optimal alignment for up to twenty short (200-300 a.a.) protein sequences in a reasonable amount of time. For alignments with more or longer sequences, a heuristic approach is more practical.

Clustal (Heuristic)

A profile based progressive alignment algorithm which uses a number of heuristics to rapidly generate multiple sequence alignments, including phylogeny and scalable gap penalties.

Clustal

http://clustal.org

ClustalW and Clustal Omega

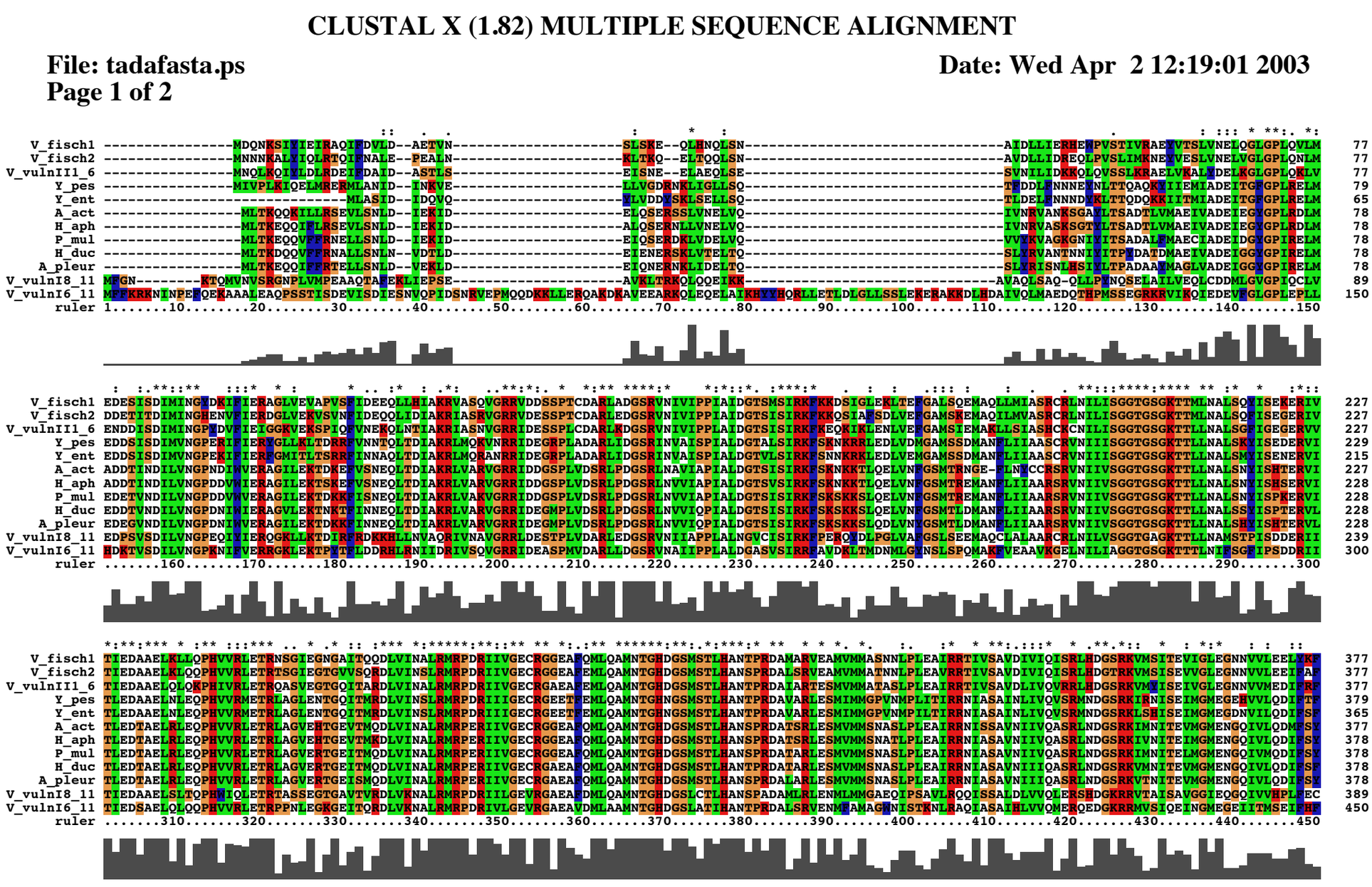
These command line tools for multiple sequence alignment first generate a pairwise distance matrix for all the sequences by pairwise dynamic programming alignment. ClustalW then estimates evolutionary distance from similarity scores and constructs a guide tree using the neighbor joining distance matrix method. Dynamic programming is then used to align the most closely related pairs of sequences. A sequence profile is constructed from these alignments, and the remaining sequences are progressively aligned to each other in order of decreasing similarity by profile-profile, profile-sequence or sequence-sequence alignment, until a complete multiple sequence alignment has been generated. Clustal Omega uses a faster, more accurate HMM alignment engine for improved performance.

Clustal automatically chooses the optimal scoring matrix for protein alignments based on whether the sequences are close or distant neighbors in the tree. Thus it might use BLOSUM62 (optimal for close relationships) for close neighbors, and BLOSUM45 (optimal for distant relationships) for distant neighbors. Clustal also allows for scalable gap penalties in protein profile alignments. A gap opening next to a highly conserved residue can be more heavily penalized than a gap opening next to an unconserved residue, for example.

ClustalX

This is a version of ClustalW with a graphical user interface, which is more intuitive to use, though the formatting requirements for input files need to be followed closely. It can display and output multiple sequence alignments and phylogenetic trees.

Clustal X



Lecture 5: Sequence Analysis
March 6, 2017