SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF BIO AND CHEMICAL ENGINEERING
DEPARTMENT OF BIOINFORMATICS

UNIT – 1- SBIA1201 – Sequence Analysis

Sequence analysis

In bioinformatics, sequence analysis is the process of subjecting a DNA, RNA or peptide sequence to any of a wide range of analytical methods to understand its features, function, structure, or evolution. Methodologies used include sequence alignment, searches against biological databases, and others. Since the development of methods of high-throughput production of gene and protein sequences, the rate of addition of new sequences to the databases increased exponentially. Such a collection of sequences does not, by itself, increase the scientist's understanding of the biology of organisms. However, comparing these new sequences to those with known functions is a key way of understanding the biology of an organism from which the new sequence comes. Thus, sequence analysis can be used to assign function to genes and proteins by the study of the similarities between the compared sequences. Nowadays, there are many tools and techniques that provide the sequence comparisons (sequence alignment) and analyze the alignment product to understand its biology.

Sequence analysis in molecular biology includes a very wide range of relevant topics:

- The comparison of sequences in order to find similarity, often to infer if they are related (homologous)
- Identification of intrinsic features of the sequence such as active sites, post translational modification sites, gene-structures, reading frames, distributions of introns and exons and regulatory elements
- Identification of sequence differences and variations such as point mutations and single nucleotide polymorphism (SNP) in order to get the genetic marker.
- Revealing the evolution and genetic diversity of sequences and organisms
- Identification of molecular structure from sequence alone

In chemistry, sequence analysis comprises techniques used to determine the sequence of a polymer formed of several monomers (see Sequence analysis of synthetic polymers). In molecular biology and genetics, the same process is called simply "sequencing".

In marketing, sequence analysis is often used in analytical customer relationship management applications, such as NPTB models (Next Product to Buy).

In sociology, sequence methods are increasingly used to study life-course and career trajectories, patterns of organizational and national development, conversation and interaction structure, and the problem of work/family synchrony. This body of research has given rise to the emerging subfield of social sequence analysis.

The term "sequence analysis" in biology implies subjecting a DNA or peptide sequence to sequence alignment, sequence databases, repeated sequence searches, or other bioinformatics methods on a computer. Since the development of methods of high-throughput production of gene and protein sequences during the 90s, the rate of addition of new sequences to the databases increases continuously. Such a collection of sequences does not, by itself, increase the scientist's understanding of the biology of organisms. However, comparing sequences with known functions with these new sequences is one way of understanding the biology of that organism from which the new sequence comes. Thus, sequence analysis can be used to assign function to genes and proteins by the study of the similarities between the compared sequences. Nowadays there are many tools and techniques that provide the sequence comparisons (sequence alignment) and analyze the alignment product to understand the biology.

Sequence analysis in molecular biology and bioinformatics is an automated, computer-based examination of characteristic fragments, e.g. of a DNA strand. It basically includes

1. The comparison of sequences in order to find similarity and dissimilarity in compared sequences (sequence alignment)
2. Identification of gene-structures, reading frames, distributions of introns and exons and regulatory elements
3. Finding and comparing point mutations or the single nucleotide polymorphism (SNP) in organism in order to get the genetic marker.
4. Revealing the evolution and genetic diversity of organisms.
5. Function annotation of genes.

6. The comparison of sequences in order to find similarity, often to infer if they are related (homologous)
7. Identification of intrinsic features of the sequence such as active sites, post translational modification sites, gene-structures, reading frames, distributions of introns and exons and regulatory elements
8. Identification of sequence differences and variations such as point mutations and single nucleotide polymorphism (SNP) in order to get the genetic marker.
9. Revealing the evolution and genetic diversity of sequences and organisms
10. Identification of molecular structure from sequence alone

Methodology

For sequence analysis, alignment method is composed of pairwise alignment (align with two sequences) and multiple alignment (align with more than two sequence). There are several tools for alignment, including: ClustalW, PROBCONS, MUSCLE, MAFFT, DIALIGN, T-Coffee, POA, and MANGO.

In bioinformatics, a sequence alignment is a way of arranging the sequences of DNA, RNA, or protein to identify regions of similarity that may be a consequence of functional, structural, or evolutionary relationships between the sequences. Aligned sequences of nucleotide or amino acid residues are typically represented as rows within a matrix. Gaps are inserted between the residues so that identical or similar characters are aligned in successive columns.

Interpretation

If two sequences in an alignment share a common ancestor, mismatches can be interpreted as point mutations and gaps as indels (that is, insertion or deletion mutations) introduced in one or both lineages in the time since they diverged from one another. In sequence alignments of proteins, the degree of similarity between amino acids occupying a particular position in the sequence can be interpreted as a rough measure of how conserved a particular region or sequence motif is among lineages. The absence of substitutions, or the presence of only

very conservative substitutions (that is, the substitution of amino acids whose side chains have similar biochemical properties) in a particular region of the sequence, suggest [3] that this region has structural or functional importance. Although DNA and RNA nucleotide bases are more similar to each other than are amino acids, the conservation of base pairs can indicate a similar functional or structural role.

Alignment methods

Very short or very similar sequences can be aligned by hand. However, most interesting problems require the alignment of lengthy, highly variable or extremely numerous sequences that cannot be aligned solely by human effort. Instead, human knowledge is applied in constructing algorithms to produce high-quality sequence alignments, and occasionally in adjusting the final results to reflect patterns that are difficult to represent algorithmically (especially in the case of nucleotide sequences).

Computational approaches to sequence alignment generally fall into two categories: *global alignments* and *local alignments*. Calculating a global alignment is a form of global optimization that "forces" the alignment to span the entire length of all query sequences. By contrast, local alignments identify regions of similarity within long sequences that are often widely divergent overall. Local alignments are often preferable, but can be more difficult to calculate because of the additional challenge of identifying the regions of similarity.

A variety of computational algorithms have been applied to the sequence alignment problem, including slow but formally optimizing methods like dynamic programming, and efficient, but not as thorough heuristic algorithms or probabilistic methods designed for large-scale database search

Global alignment

The alignment attempts to match them to each other from end to end, even though parts of the alignment are not very convincing and is based on the assumption that in an alignment the two proteins are basically similar over the entire length of one another.

A tiny example:

```
LGPSTKDFGKISESREFDN
|    ||||   |
LNQLERSFGKINMRLEDA
```

Local alignment

An alignment that searches for segments of the two sequences that match well. There is no attempt to force entire sequences into an alignment, just those parts that appear to have good similarity, according to some criterion. Using the same sequences as above, one could get:

```
----------FGKI----------
         ||||
----------FGKI----------
```

It may seem that one should always use local alignments. However, it may be difficult to spot an overall similary, as opposed to just a domain-to-domain similarity, if one uses only local alignment. So global alignment is useful in some cases.

Global alignments, which attempt to align every residue in every sequence, are most useful when the sequences in the query set are similar and of roughly equal size. (This does not mean global alignments cannot end in gaps.) A general global alignment technique is the Needleman-Wunsch algorithm, which is based on dynamic programming.

Local alignments are more useful for dissimilar sequences that are suspected to contain regions of similarity or similar sequence motifs within their larger sequence context. The Smith-Waterman algorithm is a general local alignment method also based on dynamic programming. With sufficiently similar sequences, there is no difference between local and global alignments.

Pairwise alignment

Pairwise sequence alignment methods are used to find the best-matching piecewise (local) or global alignments of two query sequences. Pairwise alignments can only be used between two sequences at a time, but they are efficient to calculate and are often used for

methods that do not require extreme precision (such as searching a database for sequences with high similarity to a query).

The three primary methods of producing pairwise alignments are dot-matrix methods, dynamic programming, and word methods; however, multiple sequence alignment techniques can also align pairs of sequences. Although each method has its individual strengths and weaknesses, all three pairwise methods have difficulty with highly repetitive sequences of low information content - especially where the number of repetitions differ in the two sequences to be aligned. One way of quantifying the utility of a given pairwise alignment is the 'maximum unique match', or the longest subsequence that occurs in both query sequence.

Dot-matrix methods

A dot plot (aka contact plot or residue contact map) is a graphical method that allows the comparison of two biological sequences and identify regions of close similarity between them. It is a kind of recurrence plot. The dot-matrix approach, which implicitly produces a family of alignments for individual sequence regions, is qualitative and conceptually simple, though time-consuming to analyze on a large scale. In the absence of noise, it can be easy to visually identify certain sequence features—such as insertions, deletions, repeats, or inverted repeats—from a dot-matrix plot. To construct a dot-matrix plot, the two sequences are written along the top row and leftmost column of a two-dimensional matrix and a dot is placed at any point where the characters in the appropriate columns match—this is a typical recurrence plot. Some implementations vary the size or intensity of the dot depending on the degree of similarity of the two characters, to accommodate conservative substitutions. The dot plots of very closely related sequences will appear as a single line along the matrix's main diagonal.

Dot plots can also be used to assess repetitiveness in a single sequence. A sequence can be plotted against itself and regions that share significant similarities will appear as lines off the main diagonal. This effect can occur when a protein consists of multiple similar structural domains.

Dynamic programming

The technique of dynamic programming can be applied to produce global alignments via the Needleman-Wunsch algorithm, and local alignments via the Smith-Waterman algorithm. In typical usage, protein alignments use a substitution matrix to assign scores to amino-acid matches or mismatches, and a gap penalty for matching an amino acid in one sequence to a gap in the other. DNA and RNA alignments may use a scoring matrix, but in practice often simply assign a positive match score, a negative mismatch score, and a negative gap penalty.

Dynamic programming can be useful in aligning nucleotide to protein sequences, a task complicated by the need to take into account frameshift mutations (usually insertions or deletions). The framesearch method produces a series of global or local pairwise alignments between a query nucleotide sequence and a search set of protein sequences, or vice versa. The dynamic programming method is guaranteed to find an optimal alignment given a particular scoring function; however, identifying a good scoring function is often an empirical rather than a theoretical matter. Although dynamic programming is extensible to more than two sequences, it is prohibitively slow for large numbers of or extremely long sequences.

The Needleman–Wunsch algorithm performs a global alignment on two sequences (called *A* and *B* here). It is commonly used in bioinformatics to align protein or nucleotide sequences. The algorithm was published in 1970 by Saul B. Needleman and Christian D. Wunsch. The Needleman–Wunsch algorithm is an example of dynamic programming, and was the first application of dynamic programming to biological sequence comparison.

The Smith-Waterman algorithm is a well-known algorithm for performing local sequence alignment; that is, for determining similar regions between two nucleotide or protein sequences. Instead of looking at the total sequence, the Smith-Waterman algorithm compares segments of all possible lengths and optimizes the similarity measure.

GAPS

Gap penalty values are designed to reduce the score when a sequence alignment has been disturbed by indels. Typically the central elements used to measure the score of an alignment have been matches, mismatches and spaces. Another important element to measure alignment scores are gaps. A gap is a consecutive run of spaces in an alignment and are used to create

alignments that are better conformed to underlying biological models and more closely fit patterns that one expects to find in meaningful alignments. Gaps are represented as dashes on a protein/DNA sequence alignment. The length of a gap is scored by the number of indels (insertions/deletions) in the sequence alignment. In protein and DNA sequence matching, two sequences are aligned to determine if they have a segment each that is significantly similar. A local alignment score is assigned according to the quality of the matches in the alignment subtracted by penalties for gaps present within the alignment. The best gap costs to use with a given substitution matrix are determined empirically. Gap penalties are used with local alignment that match a contiguous sub-sequence of the first sequence with a contiguous sub-sequence of the second sequence. When comparing proteins, one uses a similarity matrix which assigns a score to each possible residue. The score should be positive for similar residues and negative for dissimilar residues pair.

Gap penalties are used during sequence alignment. Gap penalties contribute to the overall score of alignments, and therefore, the size of the gap penalty relative to the entries in the similarity matrix affects the alignment that is finally selected. Selecting a higher gap penalty will cause less favourable characters to be aligned, to avoid creating as many gaps. Gaps are usually penalized using a linear gap function that assigns an initial penalty for a gap opening, and an additional penalty for gap extensions which increase the gap length.

Linear gap penalty

Linear gap penalties have only parameter, d, which is a penalty per unit length of gap. This is almost always negative, so that the alignment with fewer gaps is favoured over the alignment with more gaps. Under a linear gap penalty, the overall penalty for one large gap is the same as for many small gaps.

Affine gap penalty

Some sequences are more likely to have a large gap, rather than many small gaps. For example, a biological sequence is much more likely to have one big gap of length 10, due to a single insertion or deletion event, than it is to have 10 small gaps of length 1. Affine gap penalties use a gap opening penalty, o, and a gap extension penalty, e. A gap of length l is then given a penalty o

+ (*l*-1)*e*. So that gaps are discouraged, *o* is almost always negative. Because a few large gaps are better than many small gaps, *e*, though negative, is almost always less negative than *o*, so as to encourage gap extension, rather than gap introduction.

Scoring Matrices

An amino acid scoring matrix is a two-dimensional array that associates a score with any specified pair of amino acids. A substitution or scoring matrix is used to evaluate possible matches and to choose the best match between possible matches and to choose the best match between two sequences.

In bioinformatics and evolutionary biology, a substitution matrix either describes the rate at which a character in a nucleotide sequence or a protein sequence changes to other character states over evolutionary time or it describes the log odds of finding two specific character states aligned. It is an application of a stochastic matrix. Substitution matrices are usually seen in the context of amino acid or DNA sequence alignments, where the similarity between sequences depends on their divergence time and the substitution rates as represented in the matrix.

Scoring matrices are used in three major applications in protein studies.

- They are used in searches of databases to detect sequences with stretches of similarity.
- They are essential for the generation of alignments of two or more sequences.
- They form the basis of distance measures used in one type of phylogenetic tree building.

Introduction

It is assumed that the sequences being sought have an evolutionary ancestral sequence in common with the query sequence. The best guess at the actual path of evolution is the path that requires the fewest evolutionary events. All substitutions are not equally likely and should be weighted to account for this. Insertions and deletions are less likely than substitutions and should be weighted to account for this. It is necessary to consider that the choice of search algorithm influences the sensitivity and selectivity of the search. The choice of similarity matrix determines

both the pattern and the extent of substitutions in the sequences the database search is most likely to discover.

There have been extensive studies looking at the frequencies in which amino acids substituted for each other during evolution. The studies involved carefully aligning all of the proteins in several families of proteins and then constructing phylogenetic trees for each family. Each phylogenetic tree can then be examined for the substitutions found on each branch. This can then be used to produce tables (scoring matrices) of the relative frequencies with which amino acids replace each other over a short evolutionary period. Thus a substitution matrix describes the likelihood that two residue types would mutate to each other in evolutionary time.

A substitution is more likely to occur between amino acids with similar biochemical properties. For example the hydrophobic amino acids Isoleucine(I) and valine(V) get a positive score on matrices adding weight to the likeliness that one will substitute for another. While the hydrophobic amino acid isoleucine has a negative score with the hydrophilic amino acid cystine(C) as the likeliness of this substitution occurring in the protein is far less. Thus matrices are used to estimate how well two residues of given types would match if they were aligned in a sequence alignment.

Importance of scoring matrices

- Scoring matrices appear in all analysis involving sequence comparison.
- The choice of matrix can strongly influence the outcome of the analysis.
- Scoring matrices implicitly represent a particular theory of evolution.

- Understanding theories underlying a given scoring matrix can aid in making proper choice.

DNA

What do the scores in the matrices represent?

- **Overall, the alignment program is evaluating the likelihood that an alignment is significant, rather than random**
- **Each individual score is the logarithm of the ratio:**

$$\frac{\text{probability of meaningful occurrence of a residue pair}}{\text{probability of random occurrence}}$$

LOG ODDS

Identity Matrix (Unitary Matrix)

Here a you only get a positive score for a match, and a score of -10000 for a mismatch. As such a high penalty is given for a mismatch, no substitution should be allowed, although a gap may be permitted

By taking into account only the matches and mismatches: **Unitary scoring matrix**

This can be achieved by assigning a no. to the matches and another no. to the mismatches.

	A	T	G	C
A	1			
T	-100	1		
G	-100	-100	1	
C	-100	-100	-100	1

It may work well for nucleic acid alignments but is inadequate for amino acid alignments.

Using this matrix for scoring protein alignments would mean ignoring protein structure and evolution.
Therefore, we needed improvements in the scoring methods by taking into account the likelihood of a certain change: this gave rise to various alternatives to the unitary scoring matrix.

To overcome the shortcomings of the unitary matrix, alternatives were devised.

1. One of the earliest suggestions: Matrices based on the minimum no. of bases that must be changed to convert a codon for one amino acid into a codon for another amino acid. This is called the minimum mutation distance matrix. Better at identifying distant relationships among protein sequences than unitary matrix.
   Shortcoming: though it incorporates the process of mutation but ignores the processes of selection that determine the mutations which will survive in a population.
2. Matrices based on selected physical, chemical or structural properties shared & not shared by different pairs of amino acids.
3. Matrices based on a combination of structural features of amino acids and genetic code • A problem with this approach is the inability to balance the contribution of the different properties to the positive selection of mutation and the ignoring of the different rates at which the different mutations are generated.

Therefore
It is important to keep the following considerations in mind while coming up with a scoring matrix:

- Metric of similarity between amino acid pairs
- Choice of scoring matrix in itself
- How a scoring matrix is chosen
- What model forms the basis for the construction of a specific scoring matrix.

EVOLUTIONARY DISTANCES

- The best improvement upon unitary method was achieved based on evolutionary distances.
- This approach was first used by Margaret Dayhoff when she extensively studied the frequencies in which amino acids substituted for each other during evolution. This was studied through alignments and construction of phylogenetic trees of each family.
- As a result, a table of relative frequencies with which amino acids substituted each other were obtained.
- PAM (Percent Accepted Mutations) family of scoring matrices were computed using this table combined with the relative frequencies of occurrence of amino acids in proteins.
- PAM matrices are biologically sound and PAM along with other log-odds matrices (eg. BLOSUM) are statistically accurate (calculated based on observed data).

LOG-ODDS SCORING

- **Log-odds matrix:** Each score in the matrix is the logarithm of an odds ratio.
- If replacements occurred at random,

$$\text{Odds ratio} = \frac{\text{Observed no. of times “A” replaces “B”}}{\text{Expected no. of times “A” replaces “B”}}$$

- Example of log odds matrix is BLOSUM

Types of matrices

- ☞ PAM (Dayhoff)
- ☞ BLOSSUM (Henikoff)

PAM (Point Accepted Mutation) matrix

Amino acid scoring matrices are traditionally PAM (Point Accepted Mutation) matrices which refer to various degrees of sensitivity depending on the evolutionary distance between sequence pairs. In this manner PAM40 is most sensitive for sequences 40 PAMs apart. PAM250 is for more distantly related sequences and is considered a good general matrix for protein database searching. For nucleotide sequence searching a simpler approach is used which either convert a PAM40 matrix into match/mismatch values which takes into consideration that a purine may be replaced by a purine and a pyrimidine by a pyrimidine.

e.g. The PAM 250 matrix

This is appropriate for searching for alignments of sequence that have diverged by 250 PAMs, 250 mutations per 100 amino acids of sequence. Because of back mutations and silent mutations this corresponds to sequences that are about 20 percent identical.

PAM Substitution matrices

- Closely related protein alignment
- 1 PAM = 1% change
- Log Odds: natural log of target frequency/background frequency
- PAM 120: closely related proteins
- PAM 120: closely related proteins

BLOSSUM (Blocks Substitution Matrix)

The BLOSUM matrices, also used for protein database search scoring (the default in BLASTp), are divided into statistical significance degrees which, in a way, are reminiscent of PAM distances. For example, BLOSUM64 is roughly equivalent to PAM 120. BLOSSUM

Blocks Substitution Matrix). BLOSSUM matrices are most sensitive for local alignment of related sequences. The BLOSUM matrices are therefore ideal when tying to identify an unknown nucleotide sequence.
e.g. Blosum 45 Matrix

- Distantly related protein alignment
- Functional Motifs
- maximum % sequence identity that still contributes independently to model
- BLOSUM 90: closely related proteins
- BLOSUM 30: highly divergent proteins

Differences between PAM and BLOSSUM

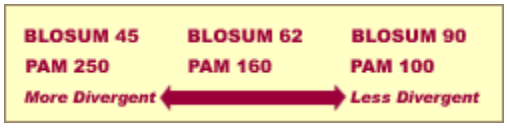
- PAM matrices are based on an explicit evolutionary model (that is, replacements are counted on the branches of a phylogenetic tree), whereas the Blosum matrices are based on an implicit rather than explicit model of evolution.

- The sequence variability in the alignments used to count replacements. The PAM matrices are based on mutations observed throughout a global alignment, this includes both highly conserved and highly mutable regions. The Blosum matrices are based only on highly conserved regions in series of alignments forbidden to contain gaps.

- The method used to count the replacements is different, unlike the PAM matrix, the Blosum procedure uses groups of sequences within which not all mutations are counted the same.

Equivalent PAM and Blossum matrices. The following matrices are roughly equivalent...

- PAM100 ==> Blosum90
- PAM120 ==> Blosum80
- PAM160 ==> Blosum60
- PAM200 ==> Blosum52
- PAM250 ==> Blosum45



Generally

- The Blosum matrices are best for detecting local alignments.
- The Blosum62 matrix is the best for detecting the majority of weak protein similarities.
- The Blosum45 matrix is the best for detecting long and weak alignments.

Summary

These 2 matrices both generally perform well, but give slightly different results. The Blosum matrices have often been the better performers, reflecting the fact that the Blosum matrices are based on the replacement patterns found in more highly conserved regions of the sequences. This seems to be an advantage as these more highly conserved regions are those discovered in database searches and they serve as anchor points in alignments involving complete sequences. It is expected that the replacements that occur in more highly conserved regions will be more restricted than those that occur in highly variable regions of the sequence.

This is supported by the different pattern of positive and negative scores in the two families of matrices. These different patterns of positive and negative scores reflect different estimates of what constitute conservative and nonconservative substitutions in the evolution of proteins. These differences reflect the differences in constructing the two families of matrices. Some of the difference is also likely to be because the Blosum matrices are based on much more data than the PAM matrices. The PAM matrices still perform quite well despite the small amount of data underlying them. The most likely reasons for this are the care used in constructing the alignments and phylogenetic trees used in counting replacements and the fact that they are based on a simple model of evolution and thus they still perform better than some of the more modern matrices that are less carefully constructed.

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF BIO AND CHEMICAL ENGINEERING
DEPARTMENT OF BIOINFORMATICS

UNIT – 2- SBIA1201 – Sequence Analysis

Similarity Search

Sequence similarity searching to identify homologous sequences is one of the first, and most informative, steps in any analysis of newly determined sequences. Modern protein sequence databases are very comprehensive, so that more than 80% of metagenomic sequence samples typically share significant similarity with proteins in sequence databases. Widely used similarity searching programs, like BLASTPSI-BLAST, SSEARCH, FASTA and the HMMER3 programs produce accurate statistical estimates, ensuring protein sequences that share significant similarity also have similar structures. Similarity searching is effective and reliable because sequences that share significant similarity can be inferred to be homologous; they share a common ancestor.

Why search databases?

- To find out if a new DNA sequence shares similarities with sequences already deposited in the databanks.
- To find proteins homologous to a putative coding ORF.
- To find similar non-coding DNA stretches in the database (for example: repeat elements, regulatory sequences).
- (To locate false priming sites for a set of PCR oligonucleotides.)

What databases are available?

- DNA (nucleotide sequences):

  The big databases: Genbank, Embl, DDBJ and their weekly updates. These databases exchange information routinely.
- Genomic databases, for example: Human, Mouse, Yeast...
- Special databases:

  ESTs (expressed sequence tags)

  STSs (sequence-tagged sites)

  EPD (eukaryotic promoter database)

  REPBASE (repetitive sequence database)

  and many others.

.

What databases are available?

- Protein (amino acid sequences):

  The big databases are:

  Uniprot-Swiss-Prot (high level of annotation)

  PIR (protein identification resource)

- Translated databases like:

  Uniprot-TREMBL (translated EMBL)

  GenPept (translation of coding regions in GenBank)

- Special databases like:

  PDB (sequences which have 3D structures)

What is a homologous sequence?

- A homologous sequence, in molecular biology, means that the sequence is similar to another sequence. The similarity is derived from common ancestry.

- Homologous proteins generally means that they are similar in their folding or their structure.

DNA vs. Protein searches

- DNA is composed of 4 characters: A,C,G,T

  It is anticipated that on the average, at least 25% of the residues of any 2 unrelated aligned sequences would be identical.
- Protein sequence is composed of 20 characters (aa). The sensitivity of the comparison is improved. It is accepted that convergence of proteins is rare, meaning that high similarity between 2 proteins almost always means homology.

DNA vs. Protein searches

- The reasons for this conclusion are:

– When comparing DNA sequences, we get significantly more random matches than we get with proteins.

– The DNA databases are much larger, and grow faster than Protein databases. Bigger databases mean more random hits!

– For DNA we usually use identity matrices, for protein more sensitive matrices like PAM and BLOSUM, which allow for better search results.

– Evolutionarily, protein sequences tend to diverge less than the DNA encoding them.

Basic principles of db searching

- When searching a database, we take a query sequence and use an algorithm (program) for the search.
- Every pair compared yields a score.
- Larger scores usually indicate a higher degree of similarity.
- A typical db search will yield a huge number of scores to be analyzed.

Specificity and sensitivity

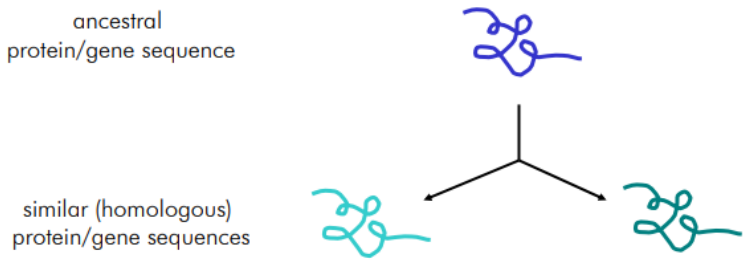
Definitions

- Sensitivity: the ability to detect "true positive" matches. The most sensitive search finds all true matches, but might have lots of false positives
- Specificity: the ability to reject "false positive" matches. The most specific search will return only true matches, but might have lots of false negatives.

Main algorithms for database searching

- FastA
  - Is theoretically better for nucleotides than blast (statistics are more rigorous)
- BLAST - Basic Local Alignment Search Tool
  - Better for proteins than for nucleotides

Importance of Similarity



similar sequences: probably have the same ancestor, share the same structure, and have a similar biological function

Importance of Similarity

Rule-of-thumb:
If your sequences are more than 100 amino acids long (or 100 nucleotides long) you can considered them as homologues if 25% of the aa are identical (70% of nucleotide for DNA). Below this value you enter the twilight zone.

Twilight zone = protein sequence similarity between ~0-20% identity:
is not statistically significant, i.e. could have arisen by chance.

Beware:
- E-value (*Expectation value*)
- length of the segments similar between the two sequences
- The number of insertions/deletions

Heuristic Sequence Alignment: Principle

- These methods are heuristic; i.e., an empirical method of computer programming in which rules of thumb are used to find solutions.
- They almost always works to find related sequences in a database search but does not have the underlying guarantee of an optimal solution like the dynamic programming algorithm.
- Advantage: This methods that are least 50-100 times faster than dynamic programming therefore better suited to search DBs.

A heuristic, or a heuristic technique, is any approach to problem solving that uses a practical method or various shortcuts in order to produce solutions that may not be optimal but are sufficient given a limited timeframe or deadline.

Heuristic Methods

A heuristic is "..a method for problem solving...often involving experimentation and trial and error.." and a heuristic algorithm is "a heuristic, is an algorithm that is able to produce an acceptable solution to a problem in many practical scenarios, but for which there is no formal proof of its correctness". Heuristics are typically used when there is no known method to find an optimal solution, under the given constraints or at all.

Why

- Heuristics are typically used when there is no known method to find an optimal solution, under the given constraints or at all.
- Allow us to incorporate knowledge about a problem or system to reduce the overall complexity of the task.
- Can help to constrain search space and/or possible solution space to avoid erroneous solutions

Assumptions for Heuristic Approaches

• Even linear time complexity is a problem for large genomes
• Databases can often be pre-processed to a degree
• Substitutions more likely than gaps
• Homologous sequences contain a lot of substitutions without gaps which can be used to help find start points in alignments

Problems

- When working with heuristic algorithms you want speed and accuracy (optimal solutions), in reality you often lose one or both
- you cannot formally prove the solution is optimal and you cannot know that the algorithm will always be fast
- do not perform well when the underlying sample is small or the problem is ill defined
- need to develop customised statistical models to go alongside the analysis to have confidence, normally randomisation based with it's associated sampling problems

First heuristic algorithms developed in sequence analysis used both heuristics and dynamic programming

- FASTA - Lipman and Pearson 1985,1988
- Clustal - Higgins et al. 1988
- BLAST - Altschul et al. 1990

Heuristics are now epidemic in Bioinformatics applied to

- classic alignment and sequence search problems
- cluster editing, partitioning problem solving
- phylogenetic parsimony
- motif detection

- protein docking
- protein structure resolution

Assessing the significance of sequence alignment

Assessing the significance of sequence alignment

- Scoring System:
  - 1. Scoring (Substitution) matrix: In proteins some mismatches are more acceptable than others. Substitution matrices give a score for each substitution of one amino-acid by another (e.g. PAM, BLOSUM)
  - 2. Gap Penalties: simulate as closely as possible the evolutionary mechanisms involved in gap occurrence. Gap opening penalty: Counted each time a gap is opened in an alignment and Gap extension penalty: Counted for each extension of a gap in an alignment.
- Based on a given scoring system: you can calculate the raw score of the alignment
  - Raw score= sum of the amino acid substitution scores and gap penalties

Assessing the significance of sequence alignment

Caveats:

1. We need a normalised score to compare different alignments, based on different scoring systems, e.g. different substitution matrices.
2. It is possible that a good long alignment gets a better raw score than a very good short alignment

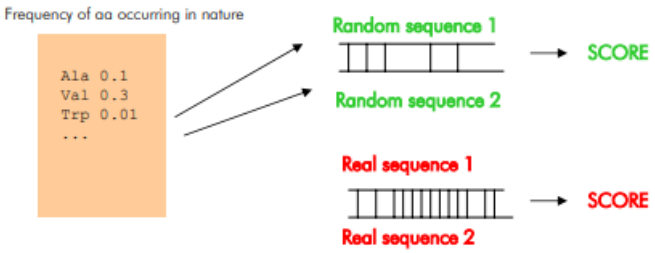
=> a method to asses the statistical significance of the alignment is needed (is an alignment biological relevant?) : E-value

Assessing the significance of sequence alignment

- How?

  ⇒ Evaluate the probability that a score between random or unrelated sequences will reach the score found between two real sequences of interest:

  If that probability is very low, the alignment score between the real sequences is significant.



If SCORE > SCORE => the alignment between the real sequences is significant

Assessing the significance of sequence alignment

In a database of size N: P x N = E

- P-value:

Probability that an alignment with this score occurs by chance in a database of size N.
The closer the P-value is towards 0, the better the alignment

- E-value:

Number of matches with this score one can expect to find by chance in a database of size N.
The closer the E-value is towards 0, the better the alignment

Word based methods

Word methods, also known as k-tuple methods, are heuristic methods that are not guaranteed to find an optimal alignment solution, but are significantly more efficient than dynamic programming. These methods are especially useful in large-scale database searches where it is understood that a large proportion of the candidate sequences will have essentially no significant

match with the query sequence. Word methods are best known for their implementation in the database search tools FASTA and the BLAST family.

Word methods identify a series of short, non-overlapping subsequences in the query sequence that is then matched to candidate database sequences. The relative positions of the word in the two sequences being compared are subtracted to obtain an offset; this will indicate a region of alignment if multiple distinct words produce the same offset. Only if this region is detected do these methods apply more sensitive alignment criteria; thus, many unnecessary comparisons with sequences of no appreciable similarity are eliminated.

BLAST

In bioinformatics, BLAST (basic local alignment search tool)[2] is an algorithm and program for comparing primary biological sequence information, such as the amino-acid sequences of proteins or the nucleotides of DNA and/or RNA sequences.

A BLAST search enables a researcher to compare a subject protein or nucleotide sequence (called a query) with a library or database of sequences, and identify database sequences that resemble the query sequence above a certain threshold. BLAST works through use of a heuristic algorithm

.
Using a heuristic method, BLAST finds homologous sequences, not by comparing either sequence in its entirety, but rather by locating short matches between the two sequences. This process of finding initial words is called seeding. It is after this first match that BLAST begins to make local alignments. While attempting to find homology in sequences, sets of common letters, known as words, are very important.

The heuristic algorithm of BLAST locates all common three-letter words between the sequence of interest and the hit sequence, or sequences, from the database. These results will then be used to build an alignment. After making words for the sequence of interest, neighborhood words are also assembled. These words must satisfy a requirement of having a score of at least the threshold, T, when compared by using a scoring matrix. Along the lines of terms stated above, if a BLASTp were being conducted, the scoring matrix that would be used would most likely be BLOSUM62. Once both words and neighborhood words are assembled and compiled, they are compared to the sequences in the database in order to find matches. The threshold score, T, determines whether a particular word will be included in the alignment or not. Once seeding has been conducted, the alignment, which is only 3 residues long, is extended in both directions by the algorithm used by BLAST. Each extension impacts the score of the alignment by either increasing or decreasing it. Should this score be higher than a pre-determined T, the alignment will be included in the results given by BLAST. However, should this score be lower than this pre-determined T, the alignment will cease to extend, preventing areas of poor alignment to be included in the BLAST results. Note, that increasing the T score limits the amount of space available to search, decreasing the number of neighborhood words, while at the same time speeding up the process of BLAST.

Algotirthm

To run the software, BLAST requires a query sequence to search for, and a sequence to search against (also called the target sequence) or a sequence database containing multiple such sequences. BLAST will find sub-sequences in the database which are similar to subsequences in the query. In typical usage, the query sequence is much smaller than the database, e.g., the query may be one thousand nucleotides while the database is several billion nucleotides.

The main idea of BLAST is that there are often High-scoring Segment Pairs (HSP) contained in a statistically significant alignment. BLAST searches for high scoring sequence alignments between the query sequence and the existing sequences in the database using a heuristic approach that approximates the Smith-Waterman algorithm. However, the exhaustive Smith-Waterman approach is too slow for searching large genomic databases such as GenBank. Therefore, the BLAST algorithm uses a heuristic approach that is less accurate than the Smith-Waterman algorithm but over 50 times faster. The speed and relatively good accuracy of BLAST are among the key technical innovations of the BLAST programs.

An overview of the BLAST algorithm (a protein to protein search) is as follows:

1. Remove low-complexity region or sequence repeats in the query sequence.
2. Make a k-letter word list of the query sequence.

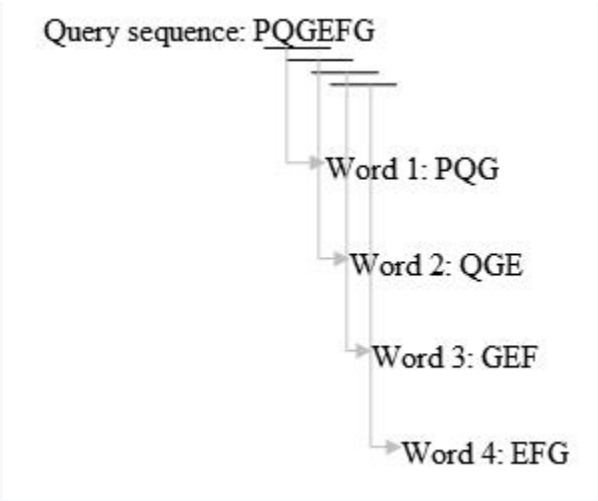


Fig. 1 The method to establish the k-letter query word list.[13]

3. Scan the database sequences for exact matches with the remaining high-scoring words.
4. Extend the exact matches to high-scoring segment pair (HSP).

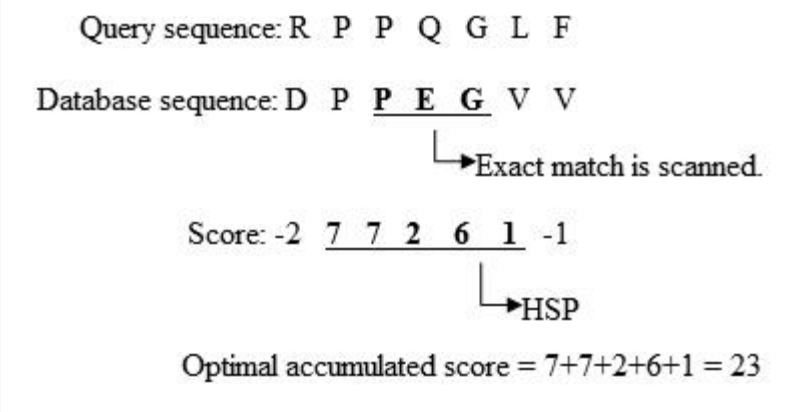


Fig. 2 The process to extend the exact match. Adapted from Biological Sequence Analysis I, Current Topics in Genome Analysis [2].

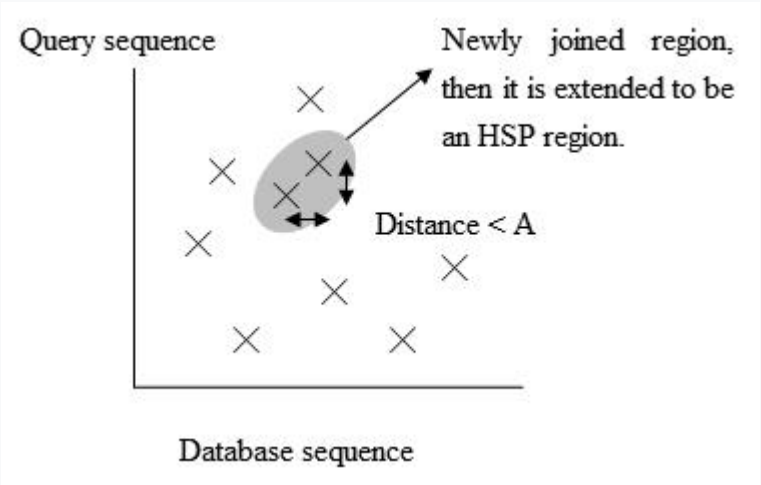


Fig. 3 The positions of the exact matches.

5. List all of the HSPs in the database whose score is high enough to be considered.
6. Evaluate the significance of the HSP score.
7. Make two or more HSP regions into a longer alignment.
8. Show the gapped Smith-Waterman local alignments of the query and each of the matched database sequences.
9. Report every match whose expect score is lower than a threshold parameter E.

BLAST is actually a family of programs (all included in the blastall executable). These include:

Nucleotide-nucleotide BLAST (blastn)

This program, given a DNA query, returns the most similar DNA sequences from the DNA database that the user specifies.

Protein-protein BLAST (blastp)

This program, given a protein query, returns the most similar protein sequences from the protein database that the user specifies.

Position-Specific Iterative BLAST (PSI-BLAST)

This program is used to find distant relatives of a protein. First, a list of all closely related proteins is created. These proteins are combined into a general profile sequence, which summarizes significant features present in these sequences. A query against the protein database is then run using this profile, and a larger group of proteins is found. This larger group is used to construct another profile, and the process is repeated. By including related proteins in the search, PSI-BLAST is much more sensitive in picking up distant evolutionary relationships than a standard protein-protein BLAST.

Nucleotide 6-frame translation-protein (blastx)

This program compares the six-frame conceptual translation products of a nucleotide query sequence (both strands) against a protein sequence database.

Nucleotide 6-frame translation-nucleotide 6-frame translation (tblastx)

This program is the slowest of the BLAST family. It translates the query nucleotide sequence in all six possible frames and compares it against the six-frame translations of a nucleotide sequence database. The purpose of tblastx is to find very distant relationships between nucleotide sequences.

Protein-nucleotide 6-frame translation (tblastn)

This program compares a protein query against the all six reading frames of a nucleotide sequence database.

Large numbers of query sequences (megablast)

When comparing large numbers of input sequences via the command-line BLAST, "megablast" is much faster than running BLAST multiple times. It concatenates many input sequences together to form a large sequence before searching the BLAST database, then post-analyze the search results to glean individual alignments and statistical values.

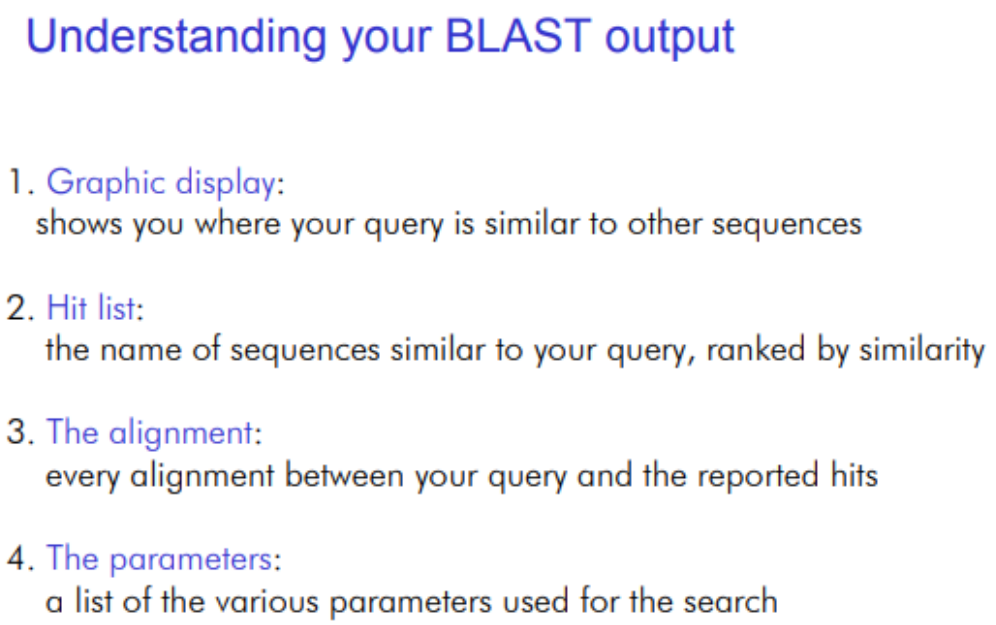
Uses of BLAST

BLAST can be used for several purposes. These include identifying species, locating domains, establishing phylogeny, DNA mapping, and comparison.

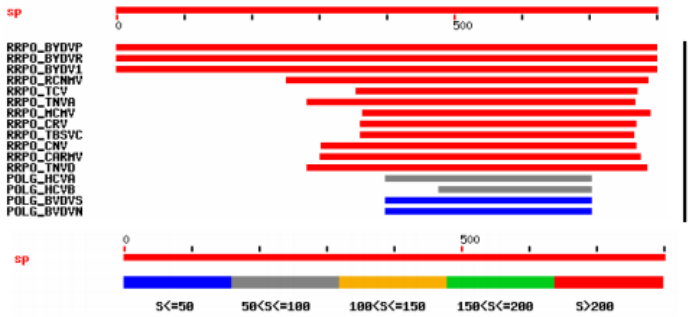
Identifying Species

With the use of BLAST, you can possibly correctly identify a species and/or find homologous species. This can be useful, for example, when you are working with a DNA sequence from an unknown species.

Locating Domains
When working with a protein sequence you can input it into BLAST, to locate known domains within the sequence of interest.
Establishing Phylogeny
Using the results received through BLAST you can create a phylogenetic tree using the BLAST web-page. It should be noted that phylogenies based on BLAST alone are less reliable than other purpose-built computational phylogenetic methods, so should only be relied upon for 'first pass' phylogenetic analyses.
DNA Mapping
When working with a known species, and looking to sequence a gene at an unknown location, BLAST can compare the chromosomal position of the sequence of interest, to relevant sequences in the database(s).
Comparison
When working with genes, BLAST can locate common genes in two related species, and can be used to map annotations from one organism to another.



Understanding your BLAST output: 1. Graphic display



query sequence

Portion of another sequence similar to your query sequence:

red, green, ochre, matches: good
grey matches: intermediate
blue: bad, (twilight zone)

The display can help you see that some matches do not extend over the entire length of your sequence => useful tool to discover domains.

Understanding your BLAST output: 2. Hit list

Sequences producing significant alignments:	Score (bits)	E Value
sp|P09505|RRPO_BYDVP Putative RNA-directed RNA polymerase (EC 2....	1652	0.0
sp|P29045|RRPO_BYDVR Putative RNA-directed RNA polymerase (EC 2....	1635	0.0
sp|P29044|RRPO_BYDV1 Putative RNA-directed RNA polymerase (EC 2....	1625	0.0
sp|P22956|RRPO_RCNMV Putative RNA-directed RNA polymerase (EC 2....	367	e-101
sp|P17460|RRPO_TCV Probable RNA-directed RNA polymerase (EC 2.7....	286	1e-76
sp|P22958|RRPO_TNVA RNA-directed RNA polymerase (EC 2.7.7.48) [C...	280	1e-74

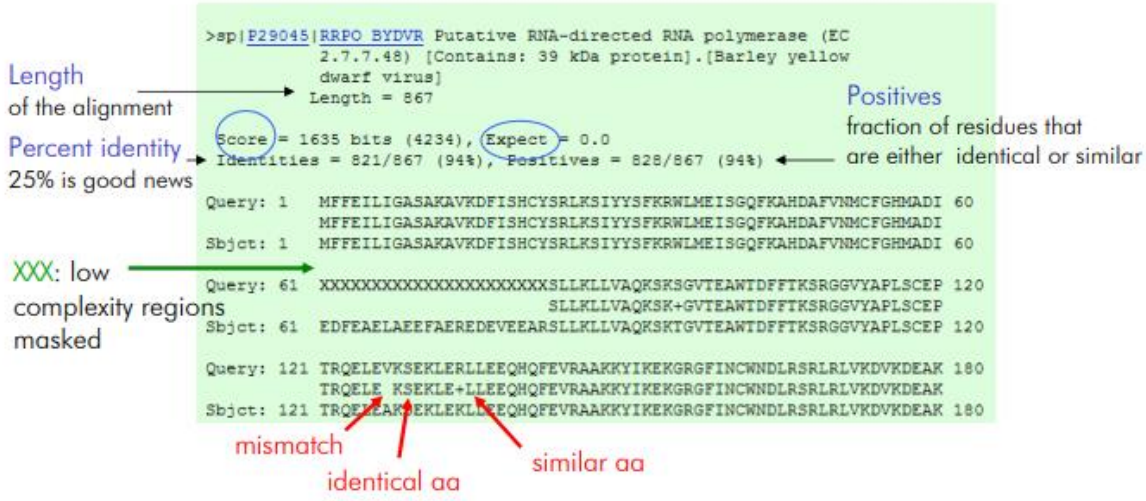
Sequence ac number and name | Description | Bit score | E-value

- Sequence ac number and name: Hyperlink to the database entry: useful annotations
- Description: better to check the full annotation

- Bit score (normalized score) : A measure of the similarity between the two sequences: the higher the better (matches below 50 bits are very unreliable)

- E-value: The lower the E-value, the better. Sequences identical to the query have an E-value of 0. Matches above 0.001 are often close to the twilight zone. As a rule-of-thumb an E-value above 10-4 (0.0001) is not necessarily interesting. If you want to be certain of the homology, your E-value must be lower than 10^{-4}

Understanding your BLAST output: 3. Alignment



A good alignment should not contain too many gaps and should have a few patches of high similarity, rather than isolated identical residues spread here and there

FASTA

It is a DNA and protein sequence alignment software package first described (as FASTP) by David J. Lipman and William R. Pearson in 1985. The original FASTP program was designed for protein sequence similarity searching. FASTA added the ability to do DNA: DNA searches, translated protein:DNA searches, and also provided a more sophisticated shuffling program for evaluating statistical significance

There are several programs in this package that allow the alignment of protein sequences and DNA sequences. Nowadays, increased computer performance makes it possible to perform searches for local alignment detection in a database using the Smith–Waterman algorithm.

FASTA is pronounced "fast A", and stands for "FAST-All", because it works with any alphabet, an extension of the original "FAST-P" (protein) and "FAST-N" (nucleotide) alignment tools.

FASTA - algorithm

FASTA takes a given nucleotide or amino acid sequence and searches a corresponding sequence database by using local sequence alignment to find matches of similar database sequences.

The FASTA program follows a largely heuristic method which contributes to the high speed of its execution. It initially observes the pattern of word hits, word-to-word matches of a given length, and marks potential matches before performing a more time-consuming optimized search using a Smith–Waterman type of algorithm.

The size taken for a word, given by the parameter kmer, controls the sensitivity and speed of the program. Increasing the k-mer value decreases number of background hits that are found. From the word hits that are returned the program looks for segments that contain a cluster of nearby hits. It then investigates these segments for a possible match.

There are some differences between fastn and fastp relating to the type of sequences used but both use four steps and calculate three scores to describe and format the sequence similarity results. These are:

- Identify regions of highest density in each sequence comparison. Taking a k-mer to equal 1 or 2.

  In this step all or a group of the identities between two sequences are found using a look up table. The k-mer value determines how many consecutive identities are required for a match to be declared. Thus the lesser the k-mer value: the more sensitive the search. k-mer=2 is frequently taken by users for protein sequences and kmer=4 or 6 for nucleotide sequences. Short oligonucleotides are usually run with k-mer= 1. The program then finds all similar **local regions**, represented as diagonals of a certain length in a dot plot, between the two sequences by counting k-mer matches and penalizing for intervening mismatches. This way, **local regions** of highest density matches in a diagonal are isolated from background hits. For protein sequences BLOSUM50 values are used for scoring k-mer matches. This ensures that groups of identities with high similarity scores contribute more to the local diagonal score than to identities with low similarity scores. Nucleotide sequences use the identity matrix for the same purpose. The best 10 local regions selected from all the diagonals put together are then saved.

- Rescan the regions taken using the scoring matrices. trimming the ends of the region to include only those contributing to the highest score.

  Rescan the 10 regions taken. This time use the relevant scoring matrix while rescoring to allow runs of identities shorter than the k-mer value. Also while rescoring conservative replacements that contribute to the similarity score are taken. Though protein sequences use the BLOSUM50 matrix, scoring matrices based on the minimum number of base changes required for a specific replacement, on identities alone, or on an alternative measure of similarity such as PAM, can also be used with the program. For each of the diagonal regions rescanned this way, a subregion with the maximum score is identified. The initial scores found in step1 are used to rank the library sequences. The highest score is referred to as *init1* score.

  - In an alignment if several initial regions with scores greater than a CUTOFF value are found, check whether the trimmed initial regions can be joined to form an approximate alignment with gaps. Calculate a similarity score that is the sum of the joined regions penalising for each gap 20 points. This initial similarity score (*initn*) is used to rank the library sequences. The score of the single best initial region found in step 2 is reported (*init1*).

  Here the program calculates an optimal alignment of initial regions as a combination of compatible regions with maximal score. This optimal alignment of initial regions can be rapidly calculated using a dynamic programming algorithm. The resulting score initn is used to rank the library sequences. This joining process increases sensitivity but decreases selectivity. A carefully calculated cut-off value is thus used to control where this step is implemented, a value that is approximately one standard deviation above the average score expected from unrelated sequences in the library. A 200-residue query sequence with k-mer 2 uses a value 28.

  - Use a banded Smith–Waterman algorithm to calculate an optimal score for alignment.

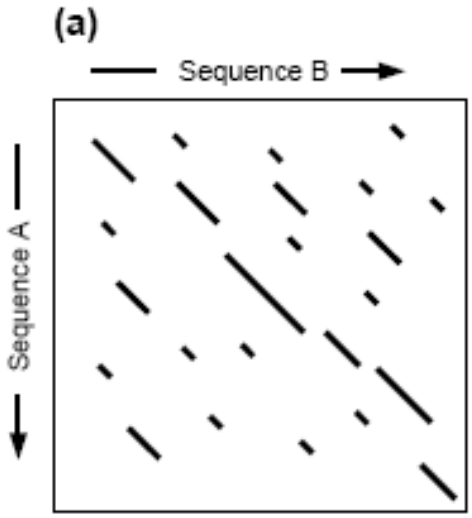
  This step uses a banded Smith–Waterman algorithm to create an optimised score (*opt*) for each alignment of query sequence to a database(library) sequence. It takes a band of 32 residues centered on the *init1* region of step2 for calculating the optimal alignment. After all sequences are searched the program plots the initial scores of each database sequence in a histogram, and calculates the statistical significance of the "opt" score. For protein sequences, the final alignment is produced using a full Smith–Waterman alignment. For DNA sequences, a banded alignment is provided.

FASTA cannot remove low complexity regions before aligning the sequences as it is possible with BLAST. This might be problematic as when the query sequence contains such regions, e.g. mini- or microsatellites repeating the same short sequence frequent times, this increases the score of not familiar sequences in the database which only match in this repeats, which occur quite frequently. Therefore, the program PRSS is added in the FASTA distribution package. PRSS shuffles the matching sequences in the database either on the one-letter level or it shuffles short segments which length the user can determine. The shuffled sequences are now aligned again and if the score is still higher than expected this is caused by the low complexity regions being mixed up still mapping to the query. By the amount of the score the shuffled sequences still attain PRSS now can predict the significance of the score of the original sequences. The higher

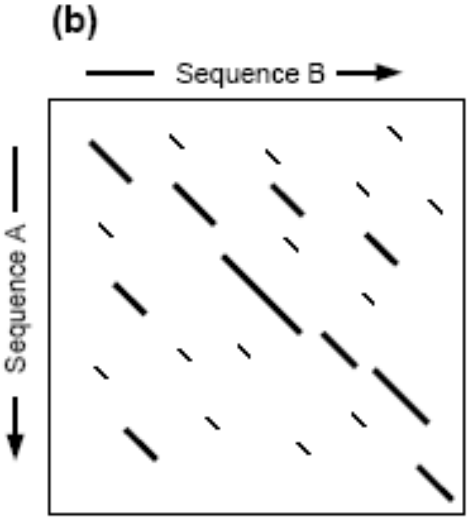
the score of the shuffled sequences the less significant the matches found between original database and query sequence.[5]

The FASTA programs find regions of local or global similarity between Protein or DNA sequences, either by searching Protein or DNA databases, or by identifying local duplications within a sequence. Other programs provide information on the statistical significance of an alignment. Like BLAST, FASTA can be used to infer functional and evolutionary relationships between sequences as well as help identify members of gene families.

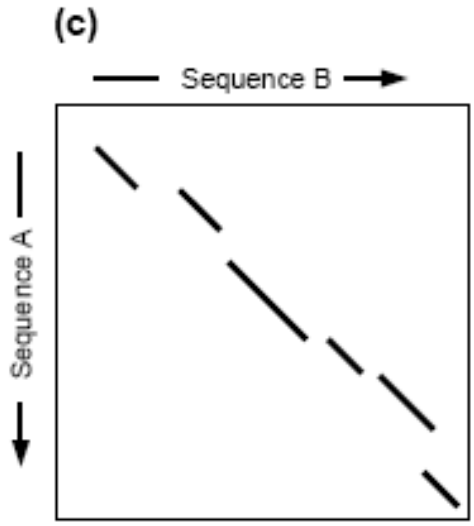
FASTA Algorithm



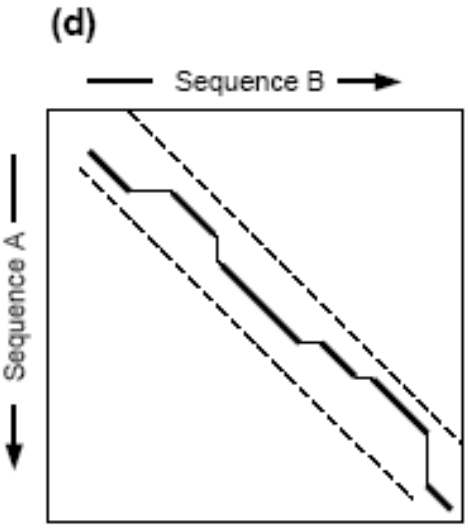
Find runs of identities



Re-score using PAM matrix
Keep top scoring segments.



Apply "joining threshold"
to eliminate segments that
are unlikely to be part of the alignment
that includes highest scoring segment.



Use dynamic programming
to optimise the alignment in a
narrow band that encompasses
the top scoring segments.

Variants of FastA

- **FastA** - Compares a DNA query sequence to a DNA database, or a protein query to a protein database, detecting the sequence type automatically. Versions 2 and 3 are in common use, version 3 having a highly improved score normalization method. It significantly reduces the overlap between the score distributions.
- **FASTX** - Compares a DNA query to a protein database. It may introduce gaps only between codons.
- **FASTY** - Compares a DNA query to a protein database, optimizing gap location, even within codons.
- **TFASTA** - Compares a protein query to a DNA database.

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF BIO AND CHEMICAL ENGINEERING
DEPARTMENT OF BIOINFORMATICS

UNIT – 3- SBIA1201 – Sequence Analysis

Multiple sequence alignment

Multiple sequence alignment (**MSA**) may refer to the process or the result of sequence alignment of three or more biological sequences, generally protein, DNA, or RNA. In many cases, the input set of query sequences are assumed to have an evolutionary relationship by which they share a linkage and are descended from a common ancestor. From the resulting MSA, sequence homology can be inferred and phylogenetic analysis can be conducted to assess the sequences' shared evolutionary origins. Visual depictions of the alignment as in the image at right illustrate mutation events such as point mutations (single amino acid or nucleotide changes) that appear as differing characters in a single alignment column, and insertion or deletion mutations (indels or gaps) that appear as hyphens in one or more of the sequences in the alignment. Multiple sequence alignment is often used to assess sequence conservation of protein domains, tertiary and secondary structures, and even individual amino acids or nucleotides.

Computational algorithms are used to produce and analyse the MSAs due to the difficulty and intractability of manually processing the sequences given their biologically-relevant length. MSAs require more sophisticated methodologies than pairwise alignment because they are more computationally complex. Most multiple sequence alignment programs use heuristic methods rather than global optimization because identifying the optimal alignment between more than a few sequences of moderate length is prohibitively computationally expensive. On the other hand, heuristic methods generally fail to give guarantees on the solution quality, with heuristic solutions shown to be often far below the optimal solution on benchmark instances

Multiple Sequence Alignment (MSA) is generally the alignment of three or more biological sequence (Protein or Nucleic acid) of similar length. From the output, homology can be inferred and the evolutionary relationship between the sequence studied.

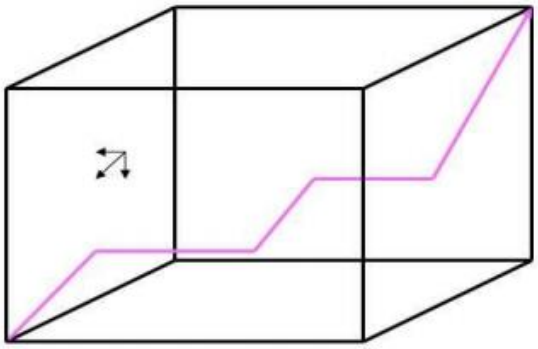
Types of MSA

- Dynamic Programming approach
- Progressive method
- Iterative method

Dynamic Programming approach

- In fact, dynamic programming is applicable to align any number of sequences.
- Computes an optimal alignment for a given score function.
- Because of its high running time, it is not typically used in practice.

Multiple Sequence Alignment Using Dynamic Programming



Finding an MSA = Finding a path through an N-Dim matrix. It's $O(L^N)$, where N is the number of sequences and L is the sequence length.

Note: More than 5 sequences takes a prohibitive amount of time.
Heuristic methods, such as those used by CLUSTAL W, are used instead.

Dynamic programming solution for multiple alignment

Recall recurrence for multiple alignment:

$$\text{Align}(S^1_i, S^2_j) = \max \begin{cases} \text{Align}(S^1_{i-1}, S^2_{j-1}) + s(a_i, a_j) \\ \text{Align}(S^1_{i-1}, S^2_j) - g \\ \text{Align}(S^1_i, S^2_{j-1}) - g \end{cases}$$

For multiple alignment, under max we have all possible combinations of matches and gaps on the last position

For k sequences dynamic programming table will have size n^k

In dynamic programming approach running time grows elementally with the number of sequences

- Two sequences $O(n^2)$
- Three sequences $O(n^3)$
- k sequences $O(n^k)$

Some approaches to accelerate computation:

- Use only part of the dynamic programming table centered along the diagonal.
- Use programming technique known as branch and bound
- Use heuristic solutions

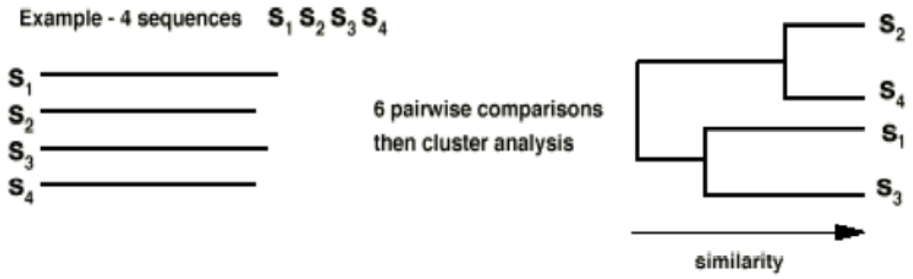
Progressive Method

- In this method, pairwise global alignment is performed for all the possible and these pairs are aligned together on the basis of their similarity.
- The most similar sequences are aligned together and thenless related sequences are added to it progressively one-by-one until a complete multiple query set is obtained.
- This method is also called hierarchical method or tree method

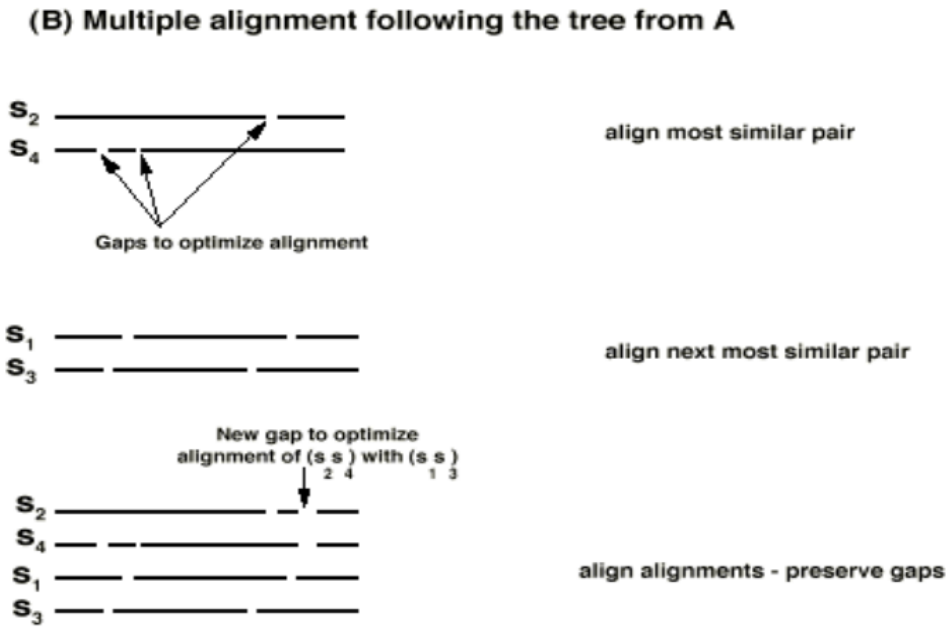
Progressive alignment (Step 1)

Steps in Multiple Alignment

(A) Pairwise Alignment



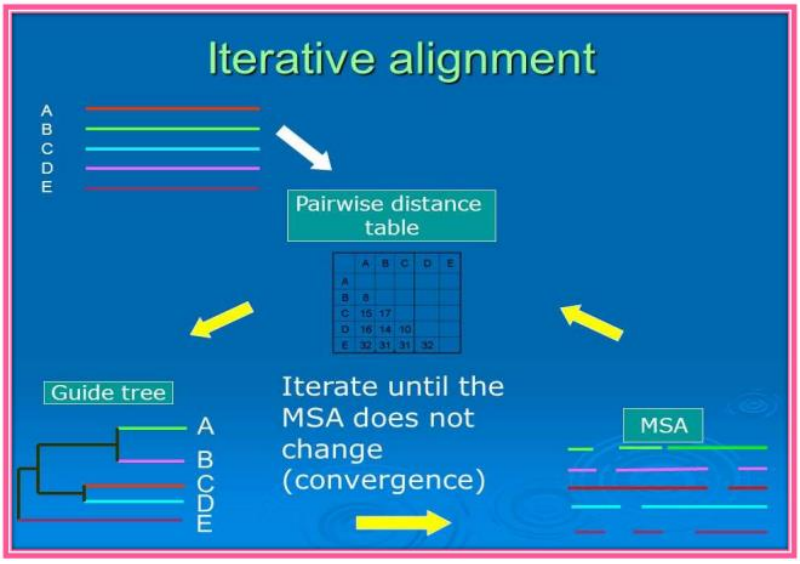
Progressive alignment (Step 2)



Iterative method

- A method of performing a series of steps to produce sucessively better approximation to align many sequences step-by-step is called iterative method.
- Here the pairwise sequence alignment is totally avoided.
- Iterative methods attempt to improve on the weak point of the progressive methods the heavy dependence on the accuracy of the initial pairwise alignment.

Steps in Iterative Method



Tools involved in MSA

- Clustal W
- Clustal W2
- Clustal Omega
- Kalign
- MAFFT
- MUSCLE
- M View
- T-Coffee
- Web PRANK
- MEME
- MACAW

Applications of MSA

- Detecting similarities between sequences(closely or distinctly related).
- Detecting conserved regions or motifs in sequences.
- Detecting of structural homologies.
- Thus, assisting the improved prediction of secondary and tertiary structures of proteins.

Multiple alignment as generalization of pairwise alignment

$S^1, S^2, \ldots, S^k$ a set of sequences over the same alphabet

As for the pair-wise alignment, the goal is to find alignment that maximizes some scoring function:

M Q P I L LP
M L R – L- P
M P V I L KP

How to score such multiple alignment?

Sum of pairs (SP) score

Example consider all pairs of letters in each column and add the scores:

$$\text{SP-score}\begin{pmatrix} A \\ V \\ V \\ - \end{pmatrix} =$$

score(A,V)+score(V,V)+score(V,-)+score(A,-)+score(A,V)

k sequences gives k(k-1)/2 addends

Remark:

Score(-,-) = 0

Sum of pairs is not prefect scoring system

No theoretical justification for the score.

- In the example below identical pairs are scored 1 and different 0.

```
A  A  A  A
A  A  A  A
A  A  A  A
A  A  A  I
A  A  I  I
A  I  I  I
----------------------
15 10 7  6
```

Entropy based score (minimum)

$$-\sum_{j}(c_j/C)\log(c_j/C)$$

c_j- number of occurrence of amino-acid j in the column

C – number of symbols in the column

```
A  A  A  A  A
A  A  A  A  I
A  A  A  A  K
A  A  A  I  L
A  A  I  I  S
A  I  I  I  W
----------------------
0  .44 .65 .69 1.79
```

(in the example natural ln)

Phylogenetic analysis

- Taxonomy - is the science of classification of organisms.
- Phylogeny - is the evolution of a genetically related group of organisms.
- Or: a study of relationships between collection of "things" (genes, proteins, organs..) that are derived from a common ancestor.

Phylogenetics - WHY?

- Find evolutionary ties between organisms.

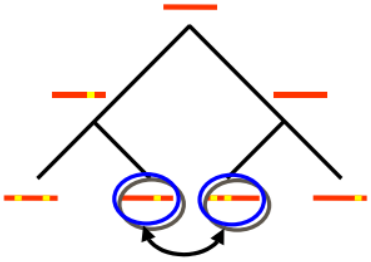
(Analyze changes occuring in different organisms during evolution).

- Find (understand) relationships between an ancestral sequence and it descendants.

(Evolution of family of sequences)

- Estimate time of divergence between a group of organisms that share a common ancestor.

Similar sequences, common ancestor...



... common ancestor, similar function

From a common ancestor sequence, two DNA sequences are diverged.

Each of these two sequences start to accumulate nucleotide substitutions.

The number of these mutations are used in molecular evolution analysis.

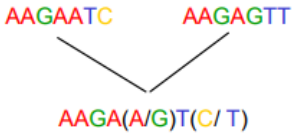
How we calculate the Degree of Divergence

- If two sequences of length N differ from each other at n sites, then their degree of divergence is:

  n/N or n/N*100%.

Relationships of Phylogenetic Analysis and Sequences Analysis

When 2 sequences found in 2 organisms are very similar, we assume that they have derived from one ancestor.



The sequences alignment reveal which positions are conserved from the ancestor sequence.

Relationships of Phylogenetic Analysis and Sequences Analysis

- The progressive multiple alignment of a group of sequences, first aligns the most similar pair.
- Then it adds the more distant pairs.
- The alignment is influenced by the "most similar" pairs and arranged accordingly, but….it does not always correctly represent the evolutionary history of the occured changes.
- Not all phylogenetic methods work this way.

Relationships of Phylogenetic Analysis and Sequences Analysis

- Most phylogenetic methods assume that each position in a sequence can change independently from the other positions.
- Gaps in alignments represent mutations in sequences such as: insertion, deletion, genetic rearrangments.
- Gaps are treated in various ways by the phylogenetic methods. Most of them ignore gaps.

Relationships of Phylogenetic Analysis and Sequences Analysis

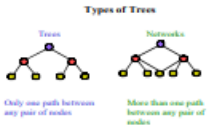
- Another approach to treat gaps is by using sequences similarity scores as the base for the phylogenetic analysis, instead of using the alignment itself, and trying to decide what happened at each position.
- The similarity scores based on scoring matrices (with gaps scores) are used by the DISTANCE methods.

What is a phylogenetic tree?

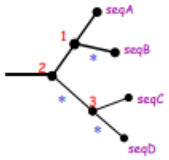
- An illustration of the evolutionary relationships among a group of organisms.
- Dendrogram is another name for a phylogenetic tree.
- A tree is composed of nodes and branches. One branch connects any two adjacent nodes. Nodes represent the taxonomic units. (sequences)

What is a phylogenetic tree?

- E.G: 2 very similar sequences will be neighbors on the outer branches and will be connected by a common internal branch.



Rooted Phylogenetic Tree

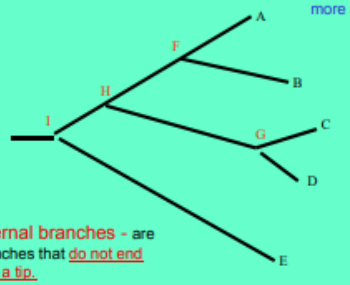


Leaves = Outer branches
Represent the taxa (sequences)

Nodes = 1 2 3
Represent the relationships
Among the taxa (sequences)
e.g Node 1 represent the ancestor
seq from which seqA and seqB derived.

Branches *
The length of the branch represent
the # of changes that occurred in the
seqs prior to the next level of separation.

External branches - are branches that end with a tip.
(FA,FB,GC,GD,IE)
more recent diversions



Internal branches - are branches that do not end with a tip.
(IH,HF,HG)
more ancient diversions

In a phylogenetic tree...

- ✓ Each NODE represents a speciation event in evolution. Beyond this point any sequence changes that occurred are specific for each branch (specie).
- ✓ The BRANCH connects 2 NODES of the tree. The length of each BRANCH between one NODE to the next, represents the # of changes that occurred until the next separation (speciation).

In a phylogenetic tree...

- ✓ NOTE: The amount of evolutionary time that passed from the separation of the 2 sequences is not known. The phylogenetic analysis can only estimate the # of changes that occurred from the time of separation.
- ✓ After the branching event, one taxon (sequence) can undergo more mutations then the other taxon.
- ✓ Topology of a tree is the branching pattern of a tree.

Tree structure

- ♠ Terminal nodes - represent the data (e.g sequences) under comparison (A,B,C,D,E), also known as OTUs, (Operational Taxonomic Units).
- ♠ Internal nodes - represent inferred ancestral units (usually without empirical data), also known as HTUs, (Hypothetical Taxonomic Units).

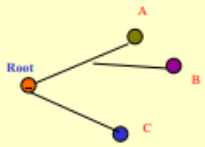
The Molecular Clock Hypothesis

- All the mutations occur in the same rate in all the tree branches.
- The rate of the mutations is the same for all positions along the sequence.

- The Molecular Clock Hypothesis is most suitable for closely related species.

Rooted Tree = Cladogram

- A phylogenetic tree that all the "objects" on it share a known common ancestor (the root).
- There exists a particular root node.
- The paths from the root to the nodes correspond to evolutionary time.



Unrooted Tree = Phenogram

A phylogenetic tree where all the "objects" on it are related descendants - but there is not enough information to specify the common ancestor (root).

- The path between nodes of the tree do not specify an evolutionary time.

Rooted versus Unrooted

- The number of tree topologies of rooted tree is much higher than that of the unrooted tree for the same number of OTUs.

- Therefore, the error of the unrooted tree topology is smaller than that of the rooted tree.

- Orthologs - genes related by speciation events. Meaning same genes in different species.

- Paralogs - genes related by duplication events. Meaning duplicated genes in the same species.

Selecting sequences for phylogenetic analysis

What *type* of sequence to use, Protein or DNA?

The rate of mutation is assumed to be the same in both coding and non-coding regions.

However, there is a difference in the substitution rate.

Selecting sequences for phylogenetic analysis

- Non-coding DNA regions have more substitution than coding regions.
- Proteins are much more conserved since they "need" to conserve their function.

So it is better to use sequences that mutate slowly (proteins) than DNA. However, if the genes are very small, or they mutate slowly, we can use them for building the trees.

Known Problems of Multiple Alignments

- Important sites could be misaligned by the software used for the sequence alignment. That will effect the significance of the site - and the tree.
  - For example: ATG as start codon, or specific amino acids in functional domains.
- Gaps - Are treated differently by different alignment programs and should play no part in building trees.

Alignment of a coding region should be compared with the alignment of their protein sequences, to be sure about the placement of gaps.

```
T  Y  R  R  S  R          ACA  TAC  AGG  CGA
                             T    Y    R    R
T  Y  R  R  S  R          ACA  TAC  AGG  CGA
                             T    Y    R    R
T  Y  R  -  S  R          ACA  TAC  AGG  ---
                             T    Y    R    -
T  Y  R  -  S  R          ACA  TAC  ---  CGA
                             T    Y    -    R
T  Y  R  R  S  R          ACA  TAC  AGG  CGA
                             T    Y    R    R
```

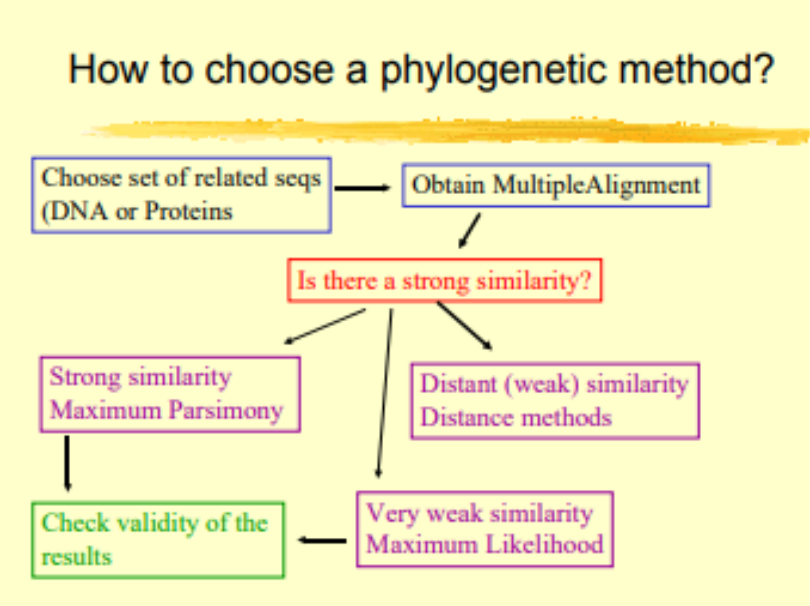
Known Problems of Multiple Alignments

- Low complexity regions - effect the multiple alignment because they create random bias for various regions of the alignment.
- Low complexity regions should be removed from the alignment before building the tree.

- If you delete these regions you need to consider the affect of the deletions on the branch lengths of the whole tree.

Selecting sequences for phylogenetic analysis

- Sequences that are being compared belong together (orthologs).
- If no ancestral sequence is available you may use an "outgroup" as a reference to measure distances. In such a case, for an outgroup you need to choose a close relative to the group being compared.
  - For example: if the group is of mammalian sequences then the outgroup should be a sequence from birds and not plants.


How to choose a phylogenetic method?
Choose set of related seqs (DNA or Proteins
Obtain MultipleAlignment
Is there a strong similarity?
Strong similarity Maximum Parsimony
Distant (weak) similarity Distance methods
Very weak similarity Maximum Likelihood
Check validity of the results

Building Phylogenetic Trees

Main methods:

- Distances matrix methods
  - Neighbour Joining, UPGMA
- Character based methods:
  - Parsimony methods
  - Maximum Likelihood method
- Validation method:
  - Bootstrapping
  - Jack Knife

Statistical Methods

✓ Bootstrapping Analysis –

Is a method for testing how good a dataset fits a evolutionary model.

This method can check the branch arrangement (topology) of a phylogenetic tree.

In Bootstrapping, the program re-samples columns in a multiple aligned group of sequences, and creates many new alignments, (with replacement the original dataset).

These new sets represent the population.

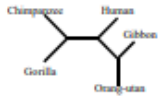
Statistical Methods

- The process is done at least 100 times.
- Phylogenetic trees are generated from all the sets.
- Part of the results will show the # of times a particular branch point occurred out of all the trees that were built.

The higher the # - the more valid the branching point.

Taken from Dr. Itai Yanai

Given the following tree, estimate the confidence of the two internal branches



Types of computational methods:

Clustering algorithms: Use pairwise distances. Are purely algorithmic methods, in which the algorithm itself defines the the tree selection criterion. Tend to be very fast programs that produce singular trees rooted by distance. No objective function to compare to other trees, even if numerous other trees could explain the data equally well. Warning: Finding a singular tree is not necessarily the same as finding the "true" evolutionary tree.

Optimality approaches: Use either character or distance data. First define an optimality criterion (minimum branch lengths, fewest number of events, highest likelihood), and then use a specific algorithm for finding trees with the best value for the objective function. Can identify many equally optimal trees, if such exist. Warning: Finding an optimal tree is not necessarily the same as finding the "true" tree.

Computational methods for finding optimal trees:

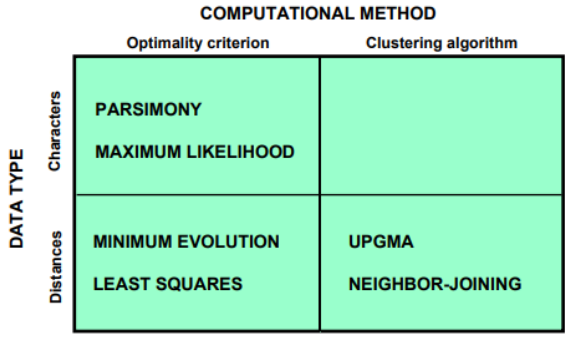
Exact algorithms: "Guarantee" to find the optimal or "best" tree for the method of choice. Two types used in tree building:

Exhaustive search: Evaluates all possible unrooted trees, choosing the one with the best score for the method.

Branch-and-bound search: Eliminates the parts of the search tree that only contain suboptimal solutions.

Heuristic algorithms: Approximate or "quick-and-dirty" methods that attempt to find the optimal tree for the method of choice, but cannot guarantee to do so. Heuristic searches often operate by "hill-climbing" methods.

Classification of phylogenetic inference methods



Parsimony methods:

Optimality criterion: The 'most-parsimonious' tree is the one that requires the fewest number of evolutionary events (*e.g.,* nucleotide substitutions, amino acid replacements) to explain the sequences.

Advantages:
- Are simple, intuitive, and logical (many possible by 'pencil-and-paper').
- Can be used on molecular *and* non-molecular (*e.g.,* morphological) data.
- Can tease apart types of similarity (shared-derived, shared-ancestral, homoplasy)
- Can be used for character (can infer the exact substitutions) and rate analysis.
- Can be used to infer the sequences of the extinct (hypothetical) ancestors.

Disadvantages:
- Are simple, intuitive, and logical (derived from "Medieval logic", not statistics!)
- Can be fooled by high levels of homoplasy ('same' events).
- Can become positively misleading in the "Felsenstein Zone":

[See Stewart (1993) for a simple explanation of parsimony analysis, and Swofford *et al.* (1996) for a detailed explanation of various parsimony methods.]

Maximum likelihood (ML) methods

Optimality criterion: ML methods evaluate phylogenetic hypotheses in terms of the probability that a proposed model of the evolutionary process and the proposed unrooted tree would give rise to the observed data. The tree found to have the highest ML value is considered to be the preferred tree.

Advantages:
- Are inherently statistical and evolutionary model-based.
- Usually the most 'consistent' of the methods available.
- Can be used for character (can infer the exact substitutions) and rate analysis.
- Can be used to infer the sequences of the extinct (hypothetical) ancestors.
- Can help account for branch-length effects in unbalanced trees.
- Can be applied to nucleotide or amino acid sequences, and other types of data.

Disadvantages:
- Are not as simple and intuitive as many other methods.
- Are computationally very intense (limits number of taxa and length of sequence).
- Like parsimony, can be fooled by high levels of homoplasy.
- Violations of the assumed model can lead to incorrect trees.

Minimum evolution (ME) methods

Optimality criterion: **The tree(s) with the shortest sum of the branch lengths (or overall tree length) is chosen as the best tree.**

Advantages:
- Can be used on indirectly-measured distances (immunological, hybridization).
- Distances can be 'corrected' for unseen events.
- Usually faster than character-based methods.
- Can be used for some rate analyses.
- Has an objective function (as compared to clustering methods).

Disadvantages:
- Information lost when characters transformed to distances.
- *Cannot* be used for character analysis.
- Slower than clustering methods.

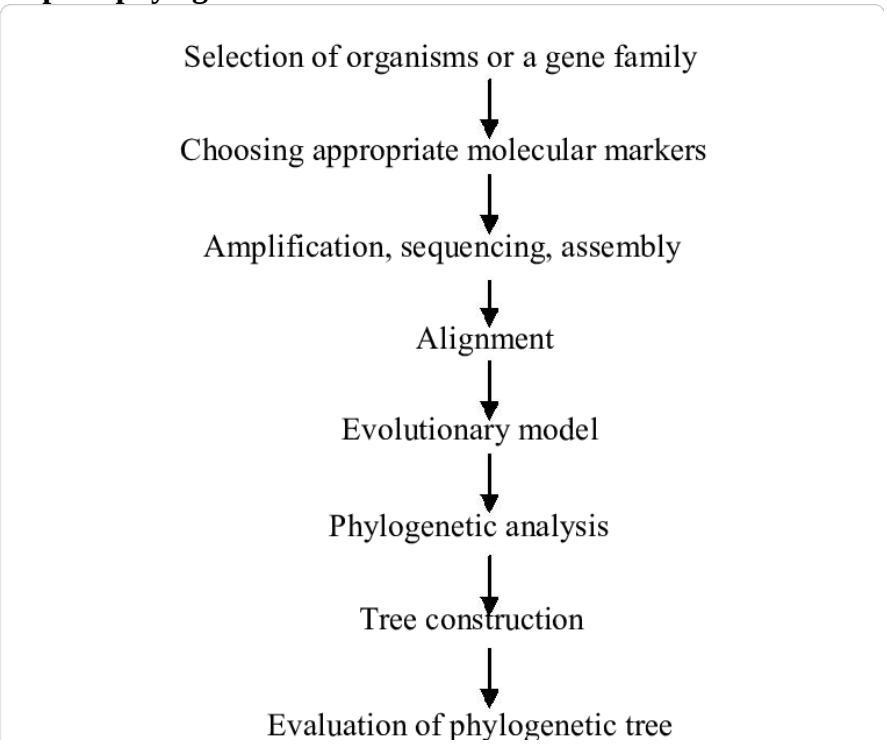
Clustering methods (UPGMA & N-J)

Optimality criterion: **NONE. The algorithm itself builds 'the' tree.**

Advantages:
- Can be used on indirectly-measured distances (immunological, hybridization).
- Distances can be 'corrected' for unseen events.
- The fastest of the methods available (N-J is screamingly fast!).
- Can therefore analyze very large datasets quickly (needed for HIV, etc.).
- Can be used for some types of rate and date analysis.

Disadvantages:
- Similarity and relationship are not necessarily the same thing, so clustering by similarity does not necessarily give an evolutionary tree.
- *Cannot* be used for character analysis!
- Have no explicit optimization criteria, so one cannot even know if the program worked properly to find the correct tree for the method.

Steps in phylogenetic tree construction



The basic steps in any phylogenetic analysis include:

1. **Assemble and align a dataset**

- The first step is to identify a protein or DNA sequence of interest and assemble a dataset consisting of other related sequences.
- DNA sequences of interest can be retrieved using NCBI BLAST or similar search tools.
- Once sequences are selected and retrieved, multiple sequence alignment is created.
- This involves arranging a set of sequences in a matrix to identify regions of homology.
- There are many websites and software programs, such as ClustalW, MSA, MAFFT, and T-Coffee, designed to perform multiple sequence on a given set of molecular data.

2. **Build (estimate) phylogenetic trees** from sequences using computational methods and stochastic models

- To build phylogenetic trees, statistical methods are applied to determine the tree topology and calculate the branch lengths that best describe the phylogenetic relationships of the aligned sequences in a dataset.
- The most common computational methods applied include distance-matrix methods, and discrete data methods, such as maximum parsimony and maximum likelihood.
- There are several software packages, such as Paup, PAML, PHYLIP, that apply these most popular methods.

3. **Statistically test and assess the estimated trees.**

- Tree estimating algorithms generate one or more optimal trees.
- This set of possible trees is subjected to a series of statistical tests to evaluate whether one tree is better than another – and if the proposed phylogeny is reasonable.
- Common methods for assessing trees include the Bootstrap and Jackknife Resampling methods, and analytical methods, such as parsimony, distance, and likelihood.

Tree evaluation: bootstrapping

- sampling technique for estimating the statistical error in situations where the underlying sampling distribution is unknown
- evaluating the reliability of the inferred tree - or better the reliability of specific branches

How to proceed:

- From the original alignment, columns in the sequence alignment are chosen at random 'sampling with replacement'
- a new alignment is constructed with the same size as the original one
- a tree is constructed

This process is repeated 100 of times**

Evaluation

Show bootstrap values on Phylogenetic trees

- majority-rule consensus tree
- map bootstrap values on the original tree
- now while evaluating from bootstrap value we are going to check a certain tree's occurrence number !!! If its 60 out of 100 times its significant, more than 50 is accountable but bellow 50 definitely rejected.

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

SCHOOL OF BIO AND CHEMICAL ENGINEERING
DEPARTMENT OF BIOINFORMATICS

UNIT – 4- SBIA1201 – Sequence Analysis

Gene structure

Gene structure is the organisation of specialised sequence elements within a gene. Genes contain the information necessary for living cells to survive and reproduce.[1][2] In most organisms, genes are made of DNA, where the particular DNA sequence determines the function of the gene. A gene is transcribed (copied) from DNA into RNA, which can either be non-coding (ncRNA) with a direct function, or an intermediate messenger (mRNA) that is then translated into protein. Each of these steps is controlled by specific sequence elements, or regions, within the gene. Every gene, therefore, requires multiple sequence elements to be functional.[2] This includes the sequence that actually encodes the functional protein or ncRNA, as well as multiple regulatory sequence regions. These regions may be as short as a few base pairs, up to many thousands of base pairs long.

Much of gene structure is broadly similar between eukaryotes and prokaryotes. These common elements largely result from the shared ancestry of cellular life in organisms over 2 billion years ago.[3] Key differences in gene structure between eukaryotes and prokaryotes reflect their divergent transcription and translation machinery.[4][5] Understanding gene structure is the foundation of understanding gene annotation, expression, and function

The structures of both eukaryotic and prokaryotic genes involve several nested sequence elements. Each element has a specific function in the multi-step process of gene expression. The sequences and lengths of these elements vary, but the same general functions are present in most genes.[2] Although DNA is a double-stranded molecule, typically only one of the strands encodes information that the RNA polymerase reads to produce protein-coding mRNA or non-coding RNA. This 'sense' or 'coding' strand, runs in the 5' to 3' direction where the numbers refer to the carbon atoms of the backbone's ribose sugar. The open reading frame (ORF) of a gene is therefore usually represented as an arrow indicating the direction in which the sense strand is read.[7]

Regulatory sequences are located at the extremities of genes. These sequence regions can either be next to the transcribed region (the promoter) or separated by many kilobases (enhancers and silencers).[8] The promoter is located at the 5' end of the gene and is composed of a core promoter sequence and a proximal promoter sequence. The core promoter marks the start site for transcription by binding RNA polymerase and other proteins necessary for copying DNA to RNA. The proximal promoter region binds transcription factors that modify the affinity of the core promoter for RNA polymerase.[9][10] Genes may be regulated by multiple enhancer and silencer sequences that further modify the activity of promoters by binding activator or repressor proteins.[11][12] Enhancers and silencers may be distantly located from the gene, many thousands of base pairs away. The binding of different transcription factors, therefore, regulates the rate of transcription initiation at different times and in different cells.[13]

Regulatory elements can overlap one another, with a section of DNA able to interact with many competing activators and repressors as well as RNA polymerase. For example, some repressor proteins can bind to the core promoter to prevent polymerase binding.[14] For genes with multiple regulatory sequences, the rate of transcription is the product of all of the elements combined.[15] Binding of activators and repressors to multiple regulatory sequences has a cooperative effect on transcription initiation.[16]

Although all organisms use both transcriptional activators and repressors, eukaryotic genes are said to be 'default off', whereas prokaryotic genes are 'default on'.[5] The core promoter of eukaryotic genes typically requires additional activation by promoter elements for expression to

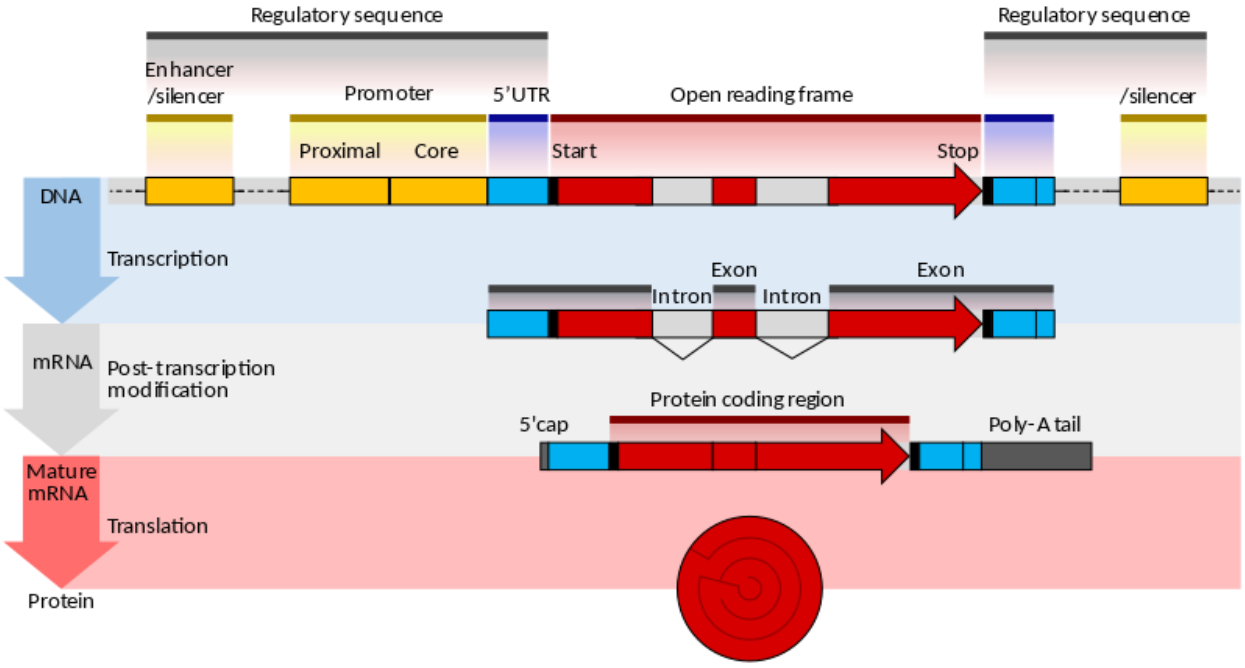
occur. The core promoter of prokaryotic genes, conversely, is sufficient for strong expression and is regulated by repressors.[5]

An additional layer of regulation occurs for protein coding genes after the mRNA has been processed to prepare it for translation to protein. Only the region between the start and stop codons encodes the final protein product. The flanking untranslated regions (UTRs) contain further regulatory sequences.[18] The 3' UTR contains a terminator sequence, which marks the endpoint for transcription and releases the RNA polymerase.[19] The 5' UTR binds the ribosome, which translates the protein-coding region into a string of amino acids that fold to form the final protein product. In the case of genes for non-coding RNAs the RNA is not translated but instead folds to be directly functional

Eukaryotes

The structure of eukaryotic genes includes features not found in prokaryotes. Most of these relate to post-transcriptional modification of pre-mRNAs to produce mature mRNA ready for translation into protein. Eukaryotic genes typically have more regulatory elements to control gene expression compared to prokaryotes.[5] This is particularly true in multicellular eukaryotes, humans for example, where gene expression varies widely among different tissues.[11]

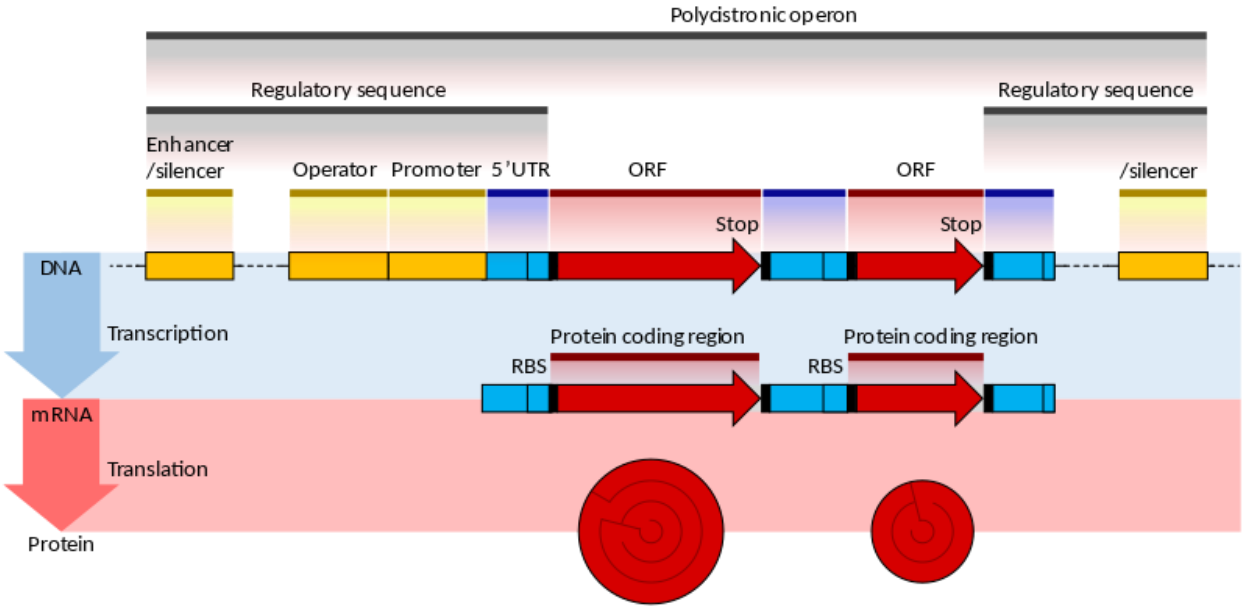
A key feature of the structure of eukaryotic genes is that their transcripts are typically subdivided into exon and intron regions. Exon regions are retained in the final mature mRNA molecule, while intron regions are spliced out (excised) during post-transcriptional processing.[22] Indeed, the intron regions of a gene can be considerably longer than the exon regions. Once spliced together, the exons form a single continuous protein-coding regions, and the splice boundaries are not detectable. Eukaryotic post-transcriptional processing also adds a 5' cap to the start of the mRNA and a poly-adenosine tail to the end of the mRNA. These additions stabilise the mRNA and direct its transport from the nucleus to the cytoplasm, although neither of these features are directly encoded in the structure of a gene.



Prokaryotes

The overall organisation of prokaryotic genes is markedly different from that of the eukaryotes. The most obvious difference is that prokaryotic ORFs are often grouped into a polycistronic operon under the control of a shared set of regulatory sequences. These ORFs are all transcribed onto the same mRNA and so are co-regulated and often serve related functions.[23][24] Each ORF typically has its own ribosome binding site (RBS) so that ribosomes simultaneously translate ORFs on the same mRNA. Some operons also display translational coupling, where the translation rates of multiple ORFs within an operon are linked.[25] This can occur when the ribosome remains attached at the end of an ORF and simply translocates along to the next without the need for a new RBS.[26] Translational coupling is also observed when translation of an ORF affects the accessibility of the next RBS through changes in RNA secondary structure.[27] Having multiple ORFs on a single mRNA is only possible in prokaryotes because their transcription and translation take place at the same time and in the same subcellular location.[23][28]

The operator sequence next to the promoter is the main regulatory element in prokaryotes. Repressor proteins bound to the operator sequence physically obstructs the RNA polymerase enzyme, preventing transcription.[29][30] Riboswitches are another important regulatory sequence commonly present in prokaryotic UTRs. These sequences switch between alternative secondary structures in the RNA depending on the concentration of key metabolites. The secondary structures then either block or reveal important sequence regions such as RBSs. Introns are extremely rare in prokaryotes and therefore do not play a significant role in prokaryotic gene regulation



Regulatory sequence

A **regulatory sequence** is a segment of a nucleic acid molecule which is capable of increasing or decreasing the expression of specific genes within an organism. Regulation of gene expression is an essential feature of all living organisms and viruses.

In DNA, regulation of gene expression normally happens at the level of RNA biosynthesis (transcription), and is accomplished through the sequence-specific binding of proteins (transcription factors) that activate or inhibit transcription. Transcription factors may act as activators, repressors, or both. Repressors often act by preventing RNA polymerase from forming a productive complex with the transcriptional initiation region (promoter), while activators facilitate formation of a productive complex. Furthermore, DNA motifs have been shown to be predictive of epigenomic modifications, suggesting that transcription factors play a role in regulating the epigenome.[2]

In RNA, regulation may occur at the level of protein biosynthesis (translation), RNA cleavage, RNA splicing, or transcriptional termination. Regulatory sequences are frequently associated with messenger RNA (mRNA) molecules, where they are used to control mRNA biogenesis or translation. A variety of biological molecules may bind to the RNA to accomplish this regulation, including proteins (e.g. translational repressors and splicing factors), other RNA molecules (e.g. miRNA) and small molecules, in the case of riboswitches.

Research to find all regulatory regions in the genomes of all sorts of organisms is under way.[3] Conserved non-coding sequences often contain regulatory regions, and so they are often the subject of these analyses.

Examples

- CAAT box
- CCAAT box
- Operator (biology)
- Pribnow box
- TATA box
- SECIS element, mRNA
- Polyadenylation signals, mRNA
- A-box
- Z-box
- C-box
- E-box
- G-box

Gene prediction

In computational biology, **gene prediction** or **gene finding** refers to the process of identifying the regions of genomic DNA that encode genes. This includes protein-coding genes as well as RNA genes, but may also include prediction of other functional elements such as regulatory regions. Gene finding is one of the first and most important steps in understanding the genome of a species once it has been sequenced.

In its earliest days, "gene finding" was based on painstaking experimentation on living cells and organisms. Statistical analysis of the rates of homologous recombination of several different genes could determine their order on a certain chromosome, and information from many such experiments could be combined to create a genetic map specifying the rough location of known genes relative to each other. Today, with comprehensive genome sequence and powerful computational resources at the disposal of the research community, gene finding has been redefined as a largely computational problem.

Determining that a sequence is functional should be distinguished from determining the function of the gene or its product. Predicting the function of a gene and confirming that the gene prediction is accurate still demands *in vivo* experimentation[1] through gene knockout and other assays, although frontiers of bioinformatics research [2] are making it increasingly possible to predict the function of a gene based on its sequence alone.

Gene prediction is one of the key steps in genome annotation, following sequence assembly, the filtering of non-coding regions and repeat masking.[3]

Gene prediction is closely related to the so-called 'target search problem' investigating how DNA-binding proteins (transcription factors) locate specific binding sites within the genome.[4][5] Many aspects of structural gene prediction are based on current understanding of underlying biochemical processes in the cell such as gene transcription, translation, protein–protein interactions and regulation processes, which are subject of active research in the various omics fields such as transcriptomics, proteomics, metabolomics, and more generally structural and functional genomics.

Importance of gene prediction

- Helps to annotate large, contiguous sequences
- Aids in the identification of fundamental and essential elements of genome such as functional genes, intron, exon, splicing sites, regulatory sites, gene encoding known proteins, motifs, EST, ACR, etc.
- Distinguish between coding and non-coding regions of a genome
- Predict complete exon – intron structures of protein coding regions
- Describe individual genes in terms of their function
- It has vast application in structural genomics ,functional genomics , metabolomics, transcriptomics, proteomics, genome studies and other genetic related studies including genetics disorders detection, treatment and prevention.

Methods of gene prediction

Similarity based methods

It is a method based on sequence similarity searches.

- It is a conceptually simple approach that is based on finding similarity in gene sequences between ESTs (expressed sequence tags), proteins, or other genomes to the input genome.
- This approach is based on the assumption that functional regions (exons) are more conserved evolutionarily than nonfunctional regions (intergenic or intronic regions).
- Once there is similarity between a certain genomic region and an EST, DNA, or protein, the similarity information can be used to infer gene structure or function of that region.

- Local alignment and global alignment are two methods based on similarity searches. The most common local alignment tool is the BLAST family of programs, which detects sequence similarity to known genes, proteins, or ESTs.
- Two more types of software, PROCRUSTES and GeneWise , use global alignment of a homologous protein to translated ORFs in a genomic sequence for gene prediction.
- A new heuristic method based on pairwise genome comparison has been implemented in the software called CSTfinder.

Abinitio Prediction

It is a method based on gene structure and signal-based searches.

- It uses gene structure as a template to detect genes
- Ab initio gene predictions rely on two types of sequence information: signal sensors and content sensors.
- Signal sensors refer to short sequence motifs, such as splice sites, branch points, polypyrimidine tracts, start codons and stop codons.
- On the other hand content sensors refer to the patterns of codon usage that are unique to a species, and allow coding sequences to be distinguished from the surrounding non-coding sequences by statistical detection algorithms. Exon detection must rely on the content sensors.
- The search by this method thus relies on the major feature present in the genes.
  - Many algorithms are applied for modeling gene structure, such as Dynamic Programming, linear discriminant analysis, Linguist methods, Hidden Markov Model and Neural Network.
  - Based on these models, a great number of ab initio gene prediction programs have been developed. Some of the frequently used ones are GeneID, FGENESH, GeneParser, GlimmerM, GENSCAN etc.

Restriction site analysis

Restriction sites, or **restriction recognition sites**, are located on a DNA molecule containing specific (4-8 base pairs in length[1]) sequences of nucleotides, which are recognized by restriction enzymes. These are generally palindromic sequences[2] (because restriction enzymes usually bind as homodimers), and a particular restriction enzyme may cut the sequence between two nucleotides within its recognition site, or somewhere nearby.

Function

For example, the common restriction enzyme EcoRI recognizes the palindromic sequence GAATTC and cuts between the G and the A on both the top and bottom strands. This leaves an overhang (an end-portion of a DNA strand with no attached complement) known as a sticky end[2] on each end of AATT. The overhang can then be used to ligate in (see DNA ligase) a piece of DNA with a complementary overhang (another EcoRI-cut piece, for example).

Some restriction enzymes cut DNA at a restriction site in a manner which leaves no overhang, called a blunt end.[2] Blunt ends are much less likely to be ligated by a DNA ligase because the blunt end doesn't have the overhanging base pair that the enzyme can recognize and match with a complementary pair.[3] Sticky ends of DNA however are more likely to successfully bind with the help of a DNA ligase because of the exposed and unpaired nucleotides. For example, a sticky end trailing with AATTG is more likely to bind with a ligase than a blunt end where both the 5' and 3' DNA strands are paired. In the case of the example the AATTG would have a complementary pair of TTAAC which would reduce the functionality of the DNA ligase enzyme

Databases

Several databases exist for restriction sites and enzymes, of which the largest noncommercial database is REBASE.[5][6] Recently, it has been shown that statistically significant nullomers (i.e. short absent motifs which are highly expected to exist) in virus genomes are restriction sites indicating that viruses have probably got rid of these motifs to facilitate invasion of bacterial hosts.[7] Nullomers Database contains a comprehensive catalogue of minimal absent motifs many of which might potentially be not-yet-known restriction motifs.

Restriction enzyme

A **restriction enzyme**, **restriction endonuclease**, or ***restrictase*** is an enzyme that cleaves DNA into fragments at or near specific recognition sites within molecules known as restriction sites.[1][2][3] Restriction enzymes are one class of the broader endonuclease group of enzymes. Restriction enzymes are commonly classified into five types, which differ in their structure and whether they cut their DNA substrate at their recognition site, or if the recognition and cleavage sites are separate from one another. To cut DNA, all restriction enzymes make two incisions, once through each sugar-phosphate backbone (i.e. each strand) of the DNA double helix.

These enzymes are found in bacteria and archaea and provide a defense mechanism against invading viruses.[4][5] Inside a prokaryote, the restriction enzymes selectively cut up *foreign* DNA in a process called *restriction digestion*; meanwhile, host DNA is protected by a modification enzyme (a methyltransferase) that modifies the prokaryotic DNA and blocks cleavage. Together, these two processes form the restriction modification system.[6]

Over 3,000 restriction enzymes have been studied in detail, and more than 800 of these are available commercially.[7] These enzymes are routinely used for DNA modification in laboratories, and they are a vital tool in molecular cloning

Restriction enzymes likely evolved from a common ancestor and became widespread via horizontal gene transfer.[24][25] In addition, there is mounting evidence that restriction endonucleases evolved as a selfish genetic element

Recognition site

Restriction enzymes recognize a specific sequence of nucleotides[2] and produce a double-stranded cut in the DNA. The recognition sequences can also be classified by the number of bases in its recognition site, usually between 4 and 8 bases, and the number of bases in the sequence will determine how often the site will appear by chance in any given genome, e.g., a 4-base pair sequence would theoretically occur once every 4^4 or 256bp, 6 bases, 4^6 or 4,096bp,

and 8 bases would be 4^8 or 65,536bp.[27] Many of them are palindromic, meaning the base sequence reads the same backwards and forwards.[28] In theory, there are two types of palindromic sequences that can be possible in DNA. The *mirror-like* palindrome is similar to those found in ordinary text, in which a sequence reads the same forward and backward on a single strand of DNA, as in GTAATG. The *inverted repeat* palindrome is also a sequence that reads the same forward and backward, but the forward and backward sequences are found in complementary DNA strands (i.e., of double-stranded DNA), as in GTATAC (GTATAC being complementary to CATATG).[29] Inverted repeat palindromes are more common and have greater biological importance than mirror-like palindromes.

EcoRI digestion produces "sticky" ends,

whereas SmaI restriction enzyme cleavage produces "blunt" ends:

Recognition sequences in DNA differ for each restriction enzyme, producing differences in the length, sequence and strand orientation (5' end or 3' end) of a sticky-end "overhang" of an enzyme restriction.[30]

Different restriction enzymes that recognize the same sequence are known as neoschizomers. These often cleave in different locales of the sequence. Different enzymes that recognize and cleave in the same location are known as isoschizomers.

Types

Naturally occurring restriction endonucleases are categorized into four groups (Types I, II III, and IV) based on their composition and enzyme cofactor requirements, the nature of their target sequence, and the position of their DNA cleavage site relative to the target sequence. DNA sequence analyses of restriction enzymes however show great variations, indicating that there are more than four types. All types of enzymes recognize specific short DNA sequences and carry out the endonucleolytic cleavage of DNA to give specific fragments with terminal 5'-phosphates. They differ in their recognition sequence, subunit composition, cleavage position, and cofactor requirements, as summarised below:

- Type I enzymes (EC 3.1.21.3) cleave at sites remote from a recognition site; require both ATP and S-adenosyl-L-methionine to function; multifunctional protein with both restriction digestion and methylase (EC 2.1.1.72) activities.
- Type II enzymes (EC 3.1.21.4) cleave within or at short specific distances from a recognition site; most require magnesium; single function (restriction digestion) enzymes independent of methylase.
- Type III enzymes (EC 3.1.21.5) cleave at sites a short distance from a recognition site; require ATP (but do not hydrolyse it); S-adenosyl-L-methionine stimulates the reaction but is not required; exist as part of a complex with a modification methylase (EC 2.1.1.72).
- Type IV enzymes target modified DNA, e.g. methylated, hydroxymethylated and glucosyl-hydroxymethylated DNA
- Type V enzymes utilize guide RNAs (gRNAs)

Enzyme	Source	Recognition Sequence	Cut
EcoRI	*Escherichia coli*	5'GAATTC 3'CTTAAG	5'---G AATTC---3' 3'---CTTAA G---5'
EcoRII	*Escherichia coli*	5'CCWGG 3'GGWCC	5'--- CCWGG---3' 3'---GGWCC ---5'
BamHI	*Bacillus amyloliquefaciens*	5'GGATCC 3'CCTAGG	5'---G GATCC---3' 3'---CCTAG G---5'
HindIII	*Haemophilus influenzae*	5'AAGCTT 3'TTCGAA	5'---A AGCTT---3' 3'---TTCGA A---5'
TaqI	*Thermus aquaticus*	5'TCGA 3'AGCT	5'---T CGA---3' 3'---AGC T---5'
NotI	*Nocardia otitidis*	5'GCGGCCGC 3'CGCCGGCG	5'---GC GGCCGC---3' 3'---CGCCGG CG---5'
HinFI	*Haemophilus influenzae*	5'GANTC 3'CTNAG	5'---G ANTC---3' 3'---CTNA G---5'
Sau3AI	*Staphylococcus aureus*	5'GATC 3'CTAG	5'--- GATC---3' 3'---CTAG ---5'
PvuII*	*Proteus vulgaris*	5'CAGCTG 3'GTCGAC	5'---CAG CTG---3' 3'---GTC GAC---5'
SmaI*	*Serratia marcescens*	5'CCCGGG 3'GGGCCC	5'---CCC GGG---3' 3'---GGG CCC---5'

Enzyme	Source	Recognition Sequence	Cut
HaeIII*	*Haemophilus aegyptius*	5'GGCC 3'CCGG	5'---GG CC---3' 3'---CC GG---5'
HgaI[76]	*Haemophilus gallinarum*	5'GACGC 3'CTGCG	5'---NN NN---3' 3'---NN NN---5'
AluI*	*Arthrobacter luteus*	5'AGCT 3'TCGA	5'---AG CT---3' 3'---TC GA---5'
EcoRV*	*Escherichia coli*	5'GATATC 3'CTATAG	5'---GAT ATC---3' 3'---CTA TAG---5'
EcoP15I	*Escherichia coli*	5'CAGCAGN_{25}NN 3'GTCGTCN_{25}NN	5'---CAGCAGN_{25} NN---3' 3'---GTCGTCN_{25}NN ---5'
KpnI[77]	*Klebsiella pneumoniae*	5'GGTACC 3'CCATGG	5'---GGTAC C---3' 3'---C CATGG---5'
PstI[77]	*Providencia stuartii*	5'CTGCAG 3'GACGTC	5'---CTGCA G---3' 3'---G ACGTC---5'
SacI[77]	*Streptomyces achromogenes*	5'GAGCTC 3'CTCGAG	5'---GAGCT C---3' 3'---C TCGAG---5'
SalI[77]	*Streptomyces albus*	5'GTCGAC 3'CAGCTG	5'---G TCGAC---3' 3'---CAGCT G---5'
ScaI*[77]	*Streptomyces caespitosus*	5'AGTACT 3'TCATGA	5'---AGT ACT---3' 3'---TCA TGA---5'
SpeI	*Sphaerotilus natans*	5'ACTAGT	5'---A CTAGT---3'

Enzyme	Source	Recognition Sequence	Cut
		3'TGATCA	3'---TGATC A---5'
SphI[77]	*Streptomyces phaeochromogenes*	5'GCATGC 3'CGTACG	5'---GCATG C---3' 3'---C GTACG---5'
StuI*[78][79]	*Streptomyces tubercidicus*	5'AGGCCT 3'TCCGGA	5'---AGG CCT---3' 3'---TCC GGA---5'
XbaI[77]	*Xanthomonas badrii*	5'TCTAGA 3'AGATCT	5'---T CTAGA---3' 3'---AGATC T---5'

Key:
* = blunt ends
N = C or G or T or A
W = A or T

Tools – for restriction sites analysis

- Webcutter 2.0 *(U.S.A.)*
- WatCut *(Michael Palmer, University of Waterloo, Canada)* - provides restriction analysis coupled with where the sites are located within genes.
- Restriction Site Analysis - *(University of Massachusetts Medical School, U.S,A.)* uses H. Mangalam's TACG2 program. Provides one with considerable choice of enzymes and output format, including pseudo gel maps.

- Restriction Enzyme Picker *(G. Rocap & E. Collins, School of Oceanography, University of Washington, U.S.A.)* - finds sets of 4 commercially available restriction endonucleases which together uniquely differentiate designated sequence groups from a supplied FASTA format sequence file for use in T-RFLP.

- NEBcutter *(New England Biolabs, U.S.A.)* - provides opportunities to upload local files, choose from common vector sequences or enter GenBank accession numbers. Also includes ability to map sites in genes. After you have the restriction map for this sequence you might want to consult the New England Biolabs (U.S.A.) site: The Restriction Enzyme Database for specifics on each restriction endonuclease and its availability.

- Other restriction sites include Restriction enzyme digest of DNA, RestrictionMapper, Restriction Map, and Restriction Digest.

- Restriction Analyzer *(Vladimír Cermák, molbiotools.com)* - carry out in silico restriction analysis online. Quickly find absent and unique sites. Tabular and graphical output. Analyze restriction fragments. Simulate a gel electrophoresis.

- Restriction Comparator *(Vladimír Cermák, molbiotools.com)* - Carry out parallel in silico restriction analysis online. Compare two sequences side by side. Find distinguishing restriction sites. Visualize restriction patterns.

ORF prediction

In molecular genetics, an **open reading frame** (**ORF**) is the part of a reading frame that has the ability to be translated. An ORF is a continuous stretch of codons that begins with a start codon (usually AUG) and ends at a stop codon (usually UAA, UAG or UGA). An ATG codon (AUG in terms of RNA) within the ORF (not necessarily the first) may indicate where translation starts. The transcription termination site is located after the ORF, beyond the translation stop codon. If transcription were to cease before the stop codon, an incomplete protein would be made during translation. In eukaryotic genes with multiple exons, introns are removed and exons are then joined together after transcription to yield the final mRNA for protein translation. In the context of gene finding, the start-stop definition of an ORF therefore only applies to spliced mRNAs, not genomic DNA, since introns may contain stop codons and/or cause shifts between reading frames. An alternative definition says that an ORF is a sequence that has a length divisible by three and is bounded by stop codons. This more general definition can also be useful in the context of transcriptomics and/or metagenomics, where start and/or stop codon may not be present in the obtained sequences. Such an ORF corresponds to parts of a gene rather than the complete gene.

Biological Significance

One common use of open reading frames (ORFs) is as one piece of evidence to assist in gene prediction. Long ORFs are often used, along with other evidence, to initially identify candidate protein-coding regions or functional RNA-coding regions in a DNA sequence. The presence of an ORF does not necessarily mean that the region is always translated. For example, in a randomly generated DNA sequence with an equal percentage of each nucleotide, a stop-codon would be expected once every 21 codons. A simple gene prediction algorithm for prokaryotes might look for a start codon followed by an open reading frame that is long enough to encode a typical protein, where the codon usage of that region matches the frequency characteristic for the given organism's coding regions. Therefore, some authors say that an ORF should have a minimal length, e.g. 100 codonsor 150 codons.[5] By itself even a long open reading frame is not conclusive evidence for the presence of a gene.[5] On the other hand, it has been proven that some short ORFs (sORFs) that lack the classical hallmarks of protein-coding genes (both from ncRNAs and mRNAs) can produce functional peptides. 5'-UTR of about 50% of mammal mRNAs are known to contain one or several sORFs.[8] 64–75% of experimentally found translation initiation sites of sORFs are conserved in the genomes of human and mouse and may indicate that these elements have function. However, sORFs can often be found only in

the minor forms of mRNAs and avoid the selection; the high conservatism of initiation sites may be connected with their location inside promoters of the relevant genes. This is characteristic of SLAMF1 gene, for example

Since DNA is interpreted in groups of three nucleotides (codons), a DNA strand has three distinct reading frames. The double helix of a DNA molecule has two anti-parallel strands; with the two strands having three reading frames each, there are six possible frame translations

ORF Finder

The ORF Finder (Open Reading Frame Finder) is a graphical analysis tool which finds all open reading frames of a selectable minimum size in a user's sequence or in a sequence already in the database. This tool identifies all open reading frames using the standard or alternative genetic codes. The deduced amino acid sequence can be saved in various formats and searched against the sequence database using the basic local alignment search tool (BLAST) server. The ORF Finder should be helpful in preparing complete and accurate sequence submissions. It is also packaged with the Sequin sequence submission software (sequence analyser).

SNP identification

In genetics, a **single-nucleotide polymorphism** (**SNP** /snɪp/; plural /snɪps/) is a substitution of a single nucleotide at a specific position in the genome that is present in a sufficiently large fraction of the population (e.g. 1% or more).[1]

For example, at a specific base position in the human genome, the C nucleotide may appear in most individuals, but in a minority of individuals, the position is occupied by an A. This means that there is a SNP at this specific position, and the two possible nucleotide variations – C or A – are said to be the alleles for this specific position.

SNPs pinpoint differences in our susceptibility to a wide range of diseases (e.g. sickle-cell anemia, β-thalassemia and cystic fibrosis).[2][3][4] The severity of illness and the way the body responds to treatments are also manifestations of genetic variations caused by SNPs. For example, a single-base mutation in the APOE (apolipoprotein E) gene is associated with a lower risk for Alzheimer's disease.[5]

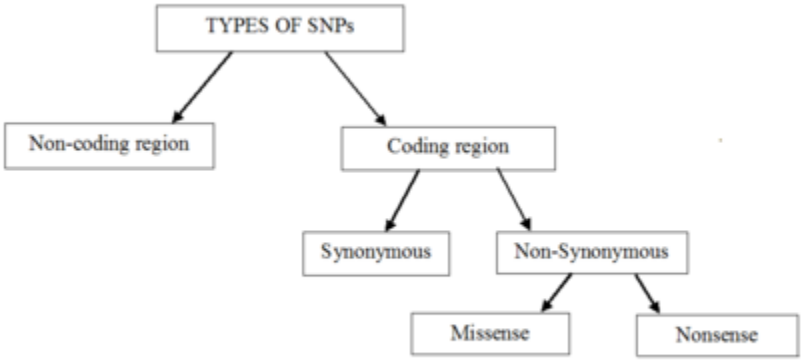
A **single-nucleotide variant** (**SNV**) is a variation in a single nucleotide without any limitations of frequency. SNVs differ from SNPs in that when a SNV is detected from one organism, the SNV could potentially be a SNP but this cannot be determined from only one organism.[6][7] SNP however means the nucleotide varies in a species' population of organisms. SNVs may arise in somatic cells which is classified as a somatic single-nucleotide variation or **single-nucleotide alteration** and can be caused by cancer.[8] SNVs also commonly arise in molecular diagnostics such as designing PCR primers to detect viruses, in which the viral RNA or DNA sample may contain SNVs.

Types

Single-nucleotide polymorphisms may fall within coding sequences of genes, non-coding regions of genes, or in the intergenic regions (regions between genes). SNPs within a coding sequence do not necessarily change the amino acid sequence of the protein that is produced, due to degeneracy of the genetic code.

SNPs in the coding region are of two types: synonymous and nonsynonymous SNPs. Synonymous SNPs do not affect the protein sequence, while nonsynonymous SNPs change the amino acid sequence of protein. The nonsynonymous SNPs are of two types: missense and nonsense.

SNPs that are not in protein-coding regions may still affect gene splicing, transcription factor binding, messenger RNA degradation, or the sequence of noncoding RNA. Gene expression affected by this type of SNP is referred to as an eSNP (expression SNP) and may be upstream or downstream from the gene.



Application

- Association studies can determine whether a genetic variant is associated with a disease or trait.[9]
- A tag SNP is a representative single-nucleotide polymorphism in a region of the genome with high linkage disequilibrium (the non-random association of alleles at two or more loci). Tag SNPs are useful in whole-genome SNP association studies, in which hundreds of thousands of SNPs across the entire genome are genotyped.
- Haplotype mapping: sets of alleles or DNA sequences can be clustered so that a single SNP can identify many linked SNPs.
- Linkage disequilibrium (LD), a term used in population genetics, indicates non-random association of alleles at two or more loci, not necessarily on the same chromosome. It refers to the phenomenon that SNP allele or DNA sequence that are close together in the genome tend to be inherited together. LD can be affected by two parameters (among other factors, such as population stratification): 1) The distance between the SNPs [the larger the distance, the lower the LD]. 2) Recombination rate [the lower the recombination rate, the higher the LD].

Frequency

More than 335 million SNPs have been found across humans from multiple populations. A typical genome differs from the reference human genome at 4 to 5 million sites, most of which (more than 99.9%) consist of SNPs and short indels.

Within a genome

The genomic distribution of SNPs is not homogenous; SNPs occur in non-coding regions more frequently than in coding regions or, in general, where natural selection is acting and "fixing" the allele (eliminating other variants) of the SNP that constitutes the most favorable genetic

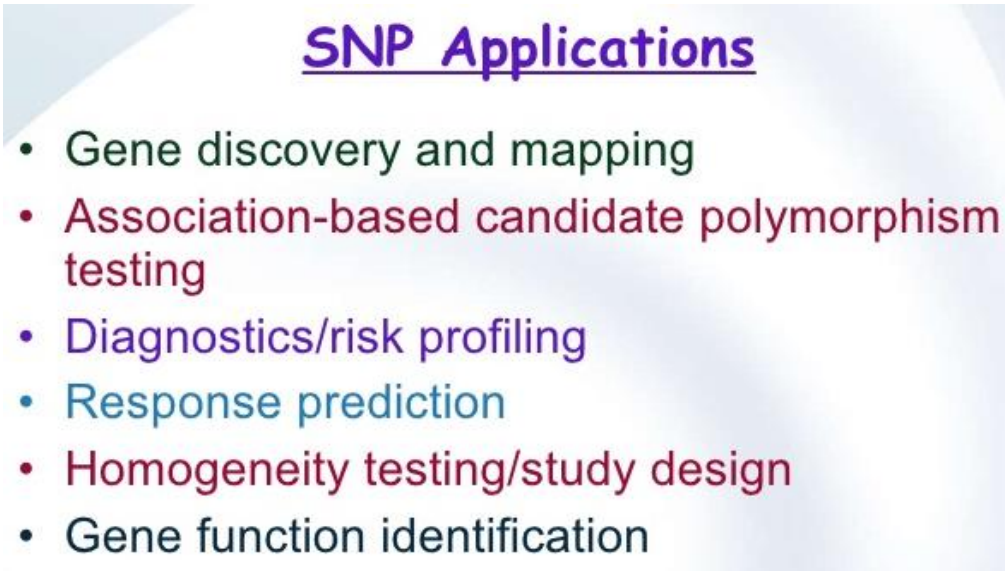
adaptation. Other factors, like genetic recombination and mutation rate, can also determine SNP density.

SNP density can be predicted by the presence of microsatellites: AT microsatellites in particular are potent predictors of SNP density, with long (AT)(n) repeat tracts tending to be found in regions of significantly reduced SNP density and low GC content.

Within a population

There are variations between human populations, so a SNP allele that is common in one geographical or ethnic group may be much rarer in another. Within a population, SNPs can be assigned a minor allele frequency—the lowest allele frequency at a locus that is observed in a particular population.[15] This is simply the lesser of the two allele frequencies for single-nucleotide polymorphisms.

With this knowledge scientists have developed new methods in analyzing population structures in less studied species. By using pooling techniques the cost of the analysis is significantly lowered. These techniques are based on sequencing a population in a pooled sample instead of sequencing every individual within the population by itself. With new bioinformatics tools there is a possibility of investigating population structure, gene flow and gene migration by observing the allele frequencies within the entire population. With these protocols there is a possibility in combining the advantages of SNPs with micro satellite markers. However, there are information lost in the process such as linkage disequilibrium and zygosity information.



Primer designing

A **primer** is a short single-stranded nucleic acid utilized by all living organisms in the initiation of DNA synthesis. DNA polymerase (responsible for DNA replication) enzymes are only capable of adding nucleotides to the 3'-end of an existing nucleic acid, requiring a primer be

bound to the template before DNA polymerase can begin a complementary strand.[1] Living organisms use solely RNA primers, while laboratory techniques in biochemistry and molecular biology that require in vitro DNA synthesis (such as DNA sequencing and polymerase chain reaction) usually use DNA primers, since they are more temperature stable.

RNA Primers

RNA primers are used by living organisms in the initiation of synthesizing a strand of DNA. A class of enzymes called primases add a complementary RNA primer to the reading template *de novo* on both the leading and lagging strands. Starting from the free 3'-OH of the primer, known as the primer terminus, a DNA polymerase can extend a newly synthesized strand. The leading strand in DNA replication is synthesized in one continuous piece moving with the replication fork, requiring only an initial RNA primer to begin synthesis. In the lagging strand, the template DNA runs in the 5′→3′ direction. Since DNA polymerase cannot add bases in the 3′→5′ direction complementary to the template strand, DNA is synthesized 'backward' in short fragments moving away from the replication fork, known as Okazaki fragments. Unlike in the leading strand, this method results in the repeated starting and stopping of DNA synthesis, requiring multiple RNA primers. Along the DNA template, primase intersperses RNA primers that DNA polymerase uses to synthesize DNA from in the 5′→3′ direction.[1]

Another example of primers being used to enable DNA synthesis is reverse transcription. Reverse transcriptase is an enzyme that uses a template strand of RNA to synthesize a complementary strand of DNA. The DNA polymerase component of reverse transcriptase requires an existing 3' end to begin synthesis.

Primer removal

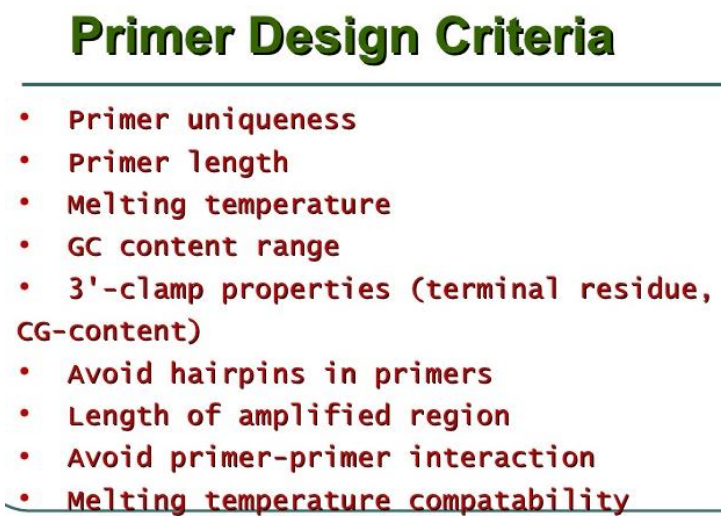
After the insertion of Okazaki fragments, the RNA primers are removed (the mechanism of removal differs between prokaryotes and eukaryotes) and replaced with new deoxyribonucleotides that fill the gaps where the RNA was present. DNA ligase then joins the fragmented strands together, completing the synthesis of the lagging strand.[1]

In prokaryotes, DNA polymerase I synthesizes the Okazaki fragment until it reaches the previous RNA primer. Then the enzyme simultaneously acts as a 5′→3′ exonuclease, removing primer ribonucleotides in front and adding deoxyribonucleotides behind until the region has been replaced by DNA, leaving a small gap in the DNA backbone between Okazaki fragments which is sealed by DNA ligase.

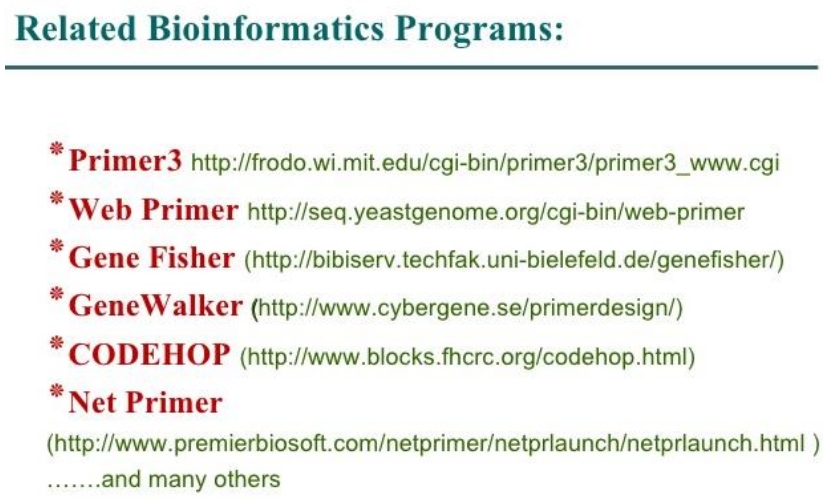
In eukaryotic primer removal, DNA polymerase δ extends the Okazaki fragment in 5′→3′ direction, and upon encountering the RNA primer from the previous Okazaki fragment, it displaces the 5′ end of the primer into a single-stranded RNA flap, which is removed by nuclease cleavage. Cleavage of the RNA flaps involves either flap structure-specific endonuclease 1 (FEN1) cleavage of short flaps, or coating of long flaps by the single-stranded DNA binding protein replication protein A (RPA) and sequential cleavage by Dna2 nuclease and FEN1

Uses of synthetic primers

Synthetic primers are chemically synthesized oligonucleotides, usually of DNA, which can be customized to anneal to a specific site on the template DNA. In solution, the primer spontaneously hybridizes with the template through Watson-Crick base pairing before being extended by DNA polymerase. The ability to create and customize synthetic primers has proven an invaluable tool necessary to a variety of molecular biological approaches involving the analysis of DNA. Both the Sanger chain termination method and the "Next-Gen" method of DNA sequencing require primers to initiate the reaction



Programs



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited "A" Grade by NAAC | 12B Status by UGC | Approved by AICTE
www.sathyabama.ac.in

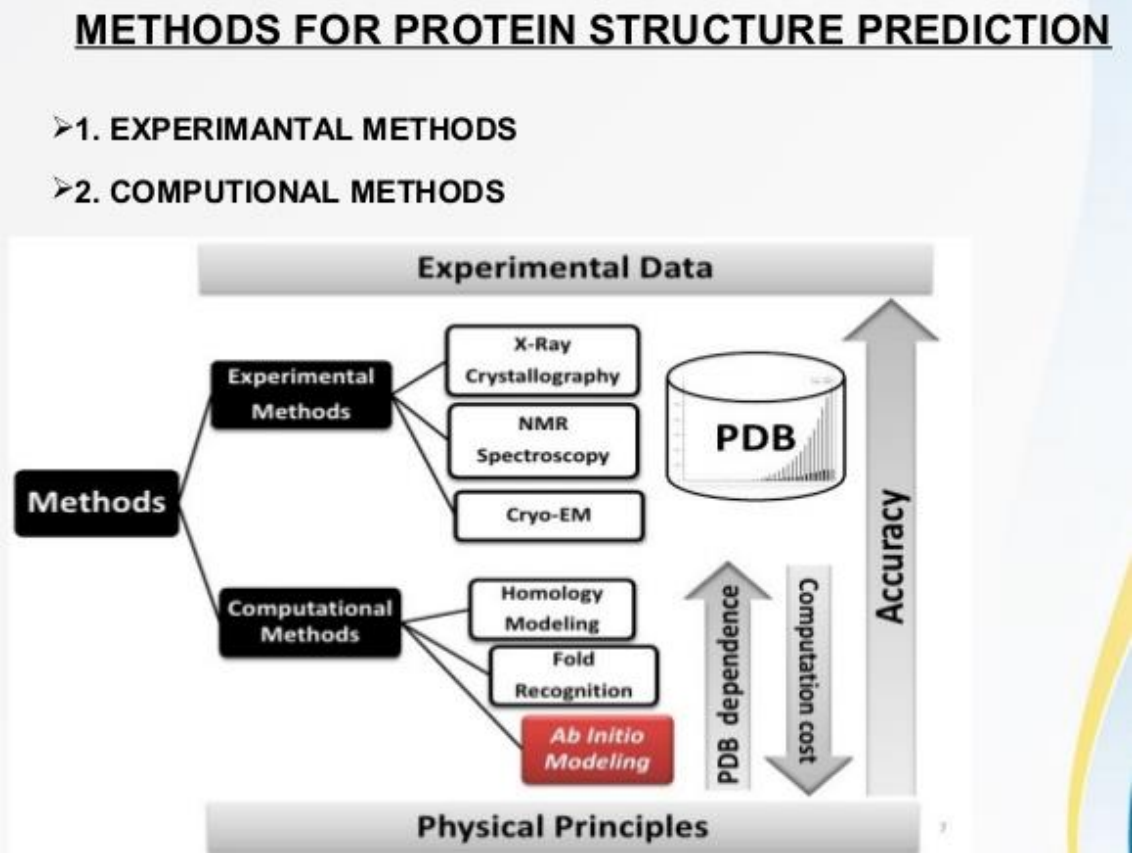
SCHOOL OF BIO AND CHEMICAL ENGINEERING
DEPARTMENT OF BIOINFORMATICS

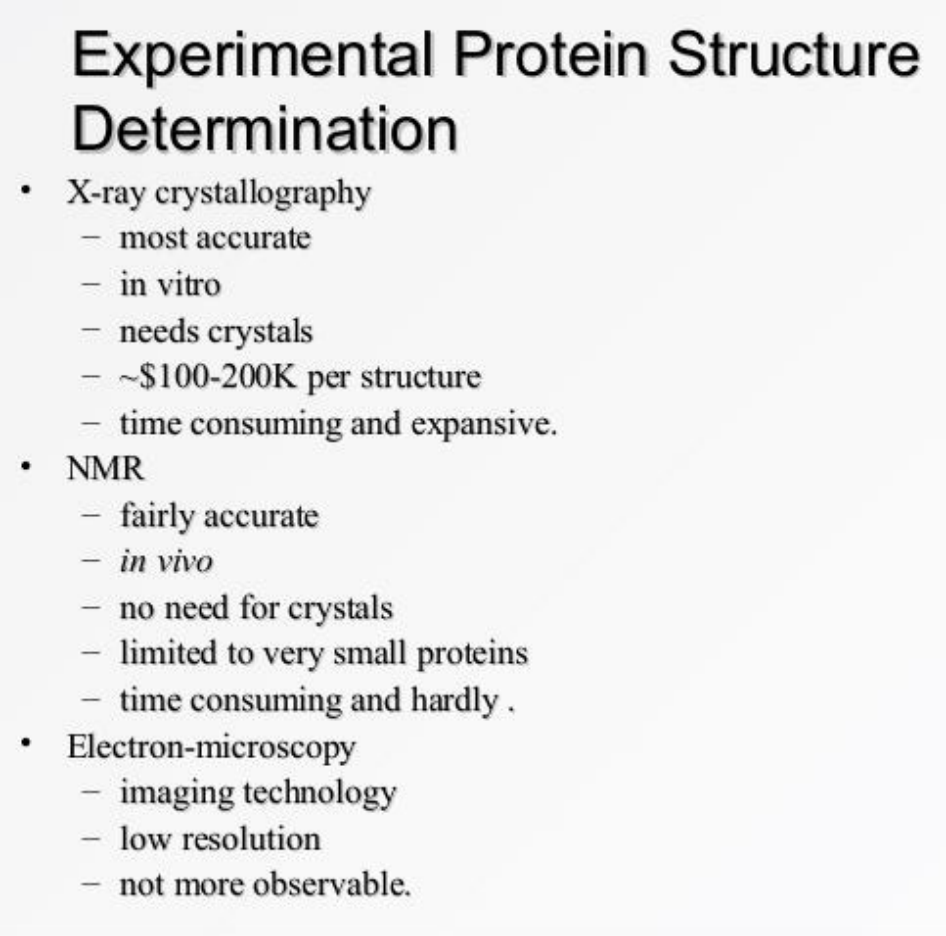
UNIT – 5- SBIA1201 – Sequence Analysis

Protein structure prediction

Protein structure prediction is the inference of the three-dimensional structure of a protein from its amino acid sequence—that is, the prediction of its secondary and tertiary structure from primary structure. Structure prediction is different from the inverse problem of protein design. Protein structure prediction is one of the most important goals pursued by computational biology; and it is important in medicine (for example, in drug design) and biotechnology (for example, in the design of novel enzymes).

Each two years, the performance of current methods is assessed in the CASP experiment (Critical Assessment of Techniques for Protein Structure Prediction). A continuous evaluation of protein structure prediction web servers is performed by the community project CAMEO3D.





Predicting Protein Structure from the Amino Acid Sequence

• Goal: Predict the 3-dimensional (tertiary) structure of a protein from the sequence of amino acids (primary structure).

• Sequence similarity methods predict secondary and tertiary structure based on homology to know proteins.

• Secondary structure predictions methods include ChouFasman, GOR, neural network, and nearest neighbor methods.

• Tertiary structure prediction methods include energy minimization, molecular dynamics, and stochastic searches of conformational space

Evolutionary Methods

Taking into account related sequences helps in identification of "structurally important" residues.

Algorithm: find similar sequences, construct multiple alignment, use alignment profile for secondary structure prediction.

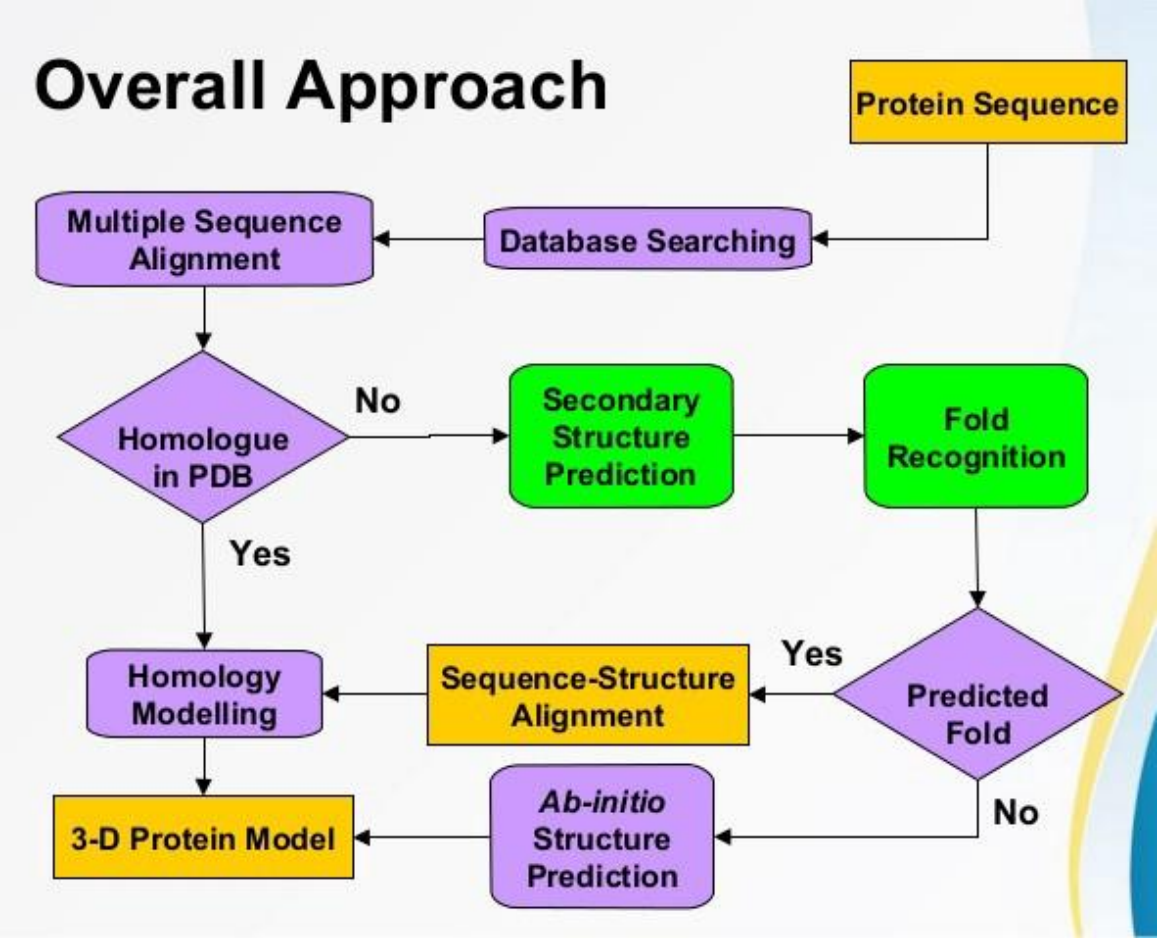
Additional information used for prediction mutation statistics residue position in sequence sequence length

Sequence similarity methods for structure prediction

• These methods can be very accurate if there is > 50% sequence similarity.

• They are rarely accurate if the sequence similarity < 30%.

• They use similar methods as used for sequence alignment such as the dynamic programming algorithm, hidden markov models, and clustering algorithms.


Overall Approach
Protein Sequence
Database Searching
Multiple Sequence Alignment
Homologue in PDB
No
Yes
Secondary Structure Prediction
Fold Recognition
Predicted Fold
Yes
No
Sequence-Structure Alignment
Homology Modelling
Ab-initio Structure Prediction
3-D Protein Model

Types of Protein Structure Predictions

- Prediction in 1D
  - secondary structure
  - solvent accessibility (which residues are exposed to water, which are buried)
  - transmembrane helices (which residues span membranes)
- Prediction in 2D
  - inter-residue/strand contacts
- Prediction in 3D
  - homology modeling
  - fold recognition (e.g. via threading)
  - *ab initio* prediction (e.g. via molecular dynamics)

Secondary structure prediction

Secondary structure prediction is a set of techniques in bioinformatics that aim to predict the secondary structures of proteins and nucleic acid sequences based only on knowledge of their primary structure. For proteins, this means predicting the formation of protein structures such as alpha helices and beta strands, while for nucleic acids it means predicting the formation of nucleic acid structures like helixes and stem-loop structures through base pairing and base stacking interactions.

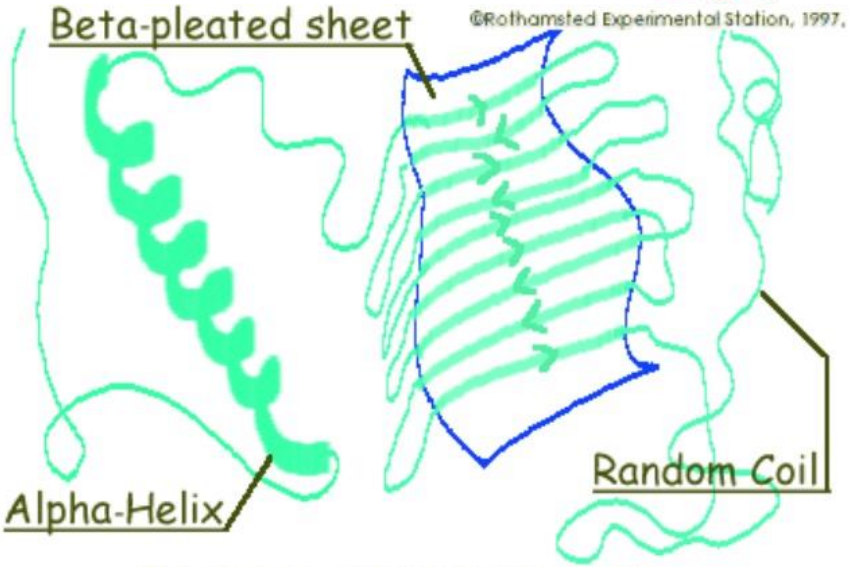
Secondary Structure

- Defined as the local conformation of protein backbone
- Primary Structure —folding— Secondary Structure
- α helix and β sheet

Secondary Structure

Regular Secondary Structure (α-helices, β-sheets)

Irregular Secondary Structure (Tight turns, Random coils, bulges)



The secondary structure is observed in a localised portion of a protein.

What is secondary structure prediction?

- Given a protein sequence (primary structure)

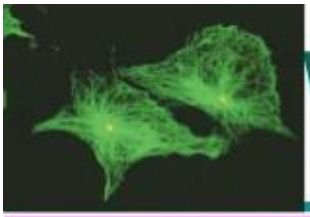
  GHWIATRGQLIREAYEDYRHFSSECPFIP

- Predict its secondary structure content

  (C=Coils H=Alpha Helix E=Beta Strands)

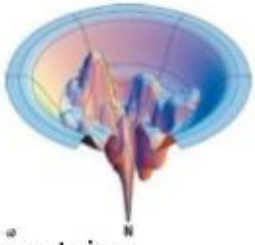
  CEEEEECHHHHHHHHHHHHCCCHHCCCCCC

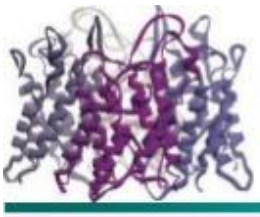
- 1st step in prediction of protein structure.
- Technique concerned with determination of secondary structure of given polypeptide by locating the Coils Alpha Helix Beta Strands in plypeptide



Why secondary structure prediction?

- secondary structure —tertiary structure prediction
- Protein function prediction
- Protein classification
- Predicting structural change
- detection and alignment of remote homology between proteins
- on detecting transmembrane regions, solvent-accessible residues, and other important features of molecules
- Detection of hydrophobic region and hydrophilic region





Prediction methods

- Statistical method
  - Chou-Fasman method, GOR I-IV
- Nearest neighbors
  - NNSSP, SSPAL
- Neural network
  - PHD, Psi-Pred, J-Pred
- Support vector machine (SVM)
- HMM

Chou-Fasman algorithm

- Chou and fasman in 1978
- It is based on assigning a set of prediction value to amino acid residue in polypeptide and applying an algorithm to the conformational parameter and positional frequency.
- conformational parameter for each amino acid is calculated by considering the relative frequency of each 20 amino acid in proteins
- By this C=Coils H=Alpha Helix E=Beta Strands are determined
- Also called preference parameter

- A table of prediction value or preference parameter for each of 20 amino acid in alpha helix ,beta plate and turn already calculated and standardised.
- To obtain the prediction value the frequency of amino acids(i) in structure is divided by of all residences in protein (s)
- i/s
- The resulting structural parameter of p(alpha),p(beta),p(turn)vary —0.5 to 1.5 for 20 amino acid

Chou-Fasman Parameters

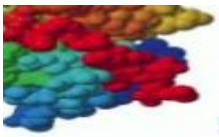
P_α		P_β		P_t	
Glu	1.51	Val	1.70	Asn	1.56
Met	1.45	Ile	1.60	Gly	1.56
Ala	1.42	Tyr	1.47	Pro	1.52
Leu	1.21	Phe	1.38	Asp	1.46
Lys	1.16	Trp	1.37	Ser	1.43
Phe	1.13	Leu	1.30	Cys	1.19
Gln	1.11	Cys	1.19	Tyr	1.14
Trp	1.08	Thr	1.19	Lys	1.01
Ile	1.08	Gln	1.10	Gln	0.98
Val	1.06	Met	1.05	Thr	0.96
Asp	1.01	Arg	0.93	Trp	0.96
His	1.00	Asn	0.89	Arg	0.95
Arg	0.98	His	0.87	His	0.95
Thr	0.83	Ala	0.83	Glu	0.74
Ser	0.77	Ser	0.75	Ala	0.66
Cys	0.70	Gly	0.75	Met	0.60
Tyr	0.69	Lys	0.74	Phe	0.60
Asn	0.67	Pro	0.55	Leu	0.59
Pro	0.57	Asp	0.54	Val	0.50
Gly	0.57	Glu	0.37	Ile	0.47

The actual algorithm contains a few simple steps:

1. Assign all of the residues in the peptide the appropriate set of parameters.
2. Scan through the peptide and identify regions where 4 out of 6 contiguous residues have P(a-helix) > 1.00. That region is declared an alpha-helix. Extend the helix in both directions until a set of four contiguous residues that have an average P(a-helix) < 1.00 is reached. That is declared the end of the helix. If the segment defined by this procedure is longer than 5 residues and the average P(a-helix) > P(b-sheet) for that segment, the segment can be assigned as a helix.
3. Repeat this procedure to locate all of the helical regions in the sequence.
4. Scan through the peptide and identify a region where 3 out of 5 of the residues have a value of P(b-sheet) > 1.00. That region is declared as a beta-sheet. Extend the sheet in both directions until a set of four contiguous residues that have an average P(b-sheet) < 1.00 is reached. That is declared the end of the beta-sheet. Any segment of the region located by this procedure is assigned as a beta-sheet if the average P(b-sheet) > 105 and the average P(b-sheet) > P(a-helix) for that region.
5. Any region containing overlapping alpha-helical and beta-sheet assignments are taken to be helical if the average P(a-helix) > P(b-sheet) for that region. It is a beta sheet if the average P(b-sheet) > P(a-helix) for that region.
6. To identify a bend at residue number j, calculate the following value

   p(t) = f(j)f(j+1)f(j+2)f(j+3)

   where the f(j+1) value for the j+1 residue is used, the f(j+2) value for the j+2 residue is used and the f(j+3) value for the j+3 residue is used. If: (1) p(t) > 0.000075; (2) the average value for P(turn) > 1.00 in the tetrapeptide; and (3) the averages for the tetrapeptide obey the inequality P(a-helix) < P(turn) > P(b-sheet), then a beta-turn is predicted at that location.

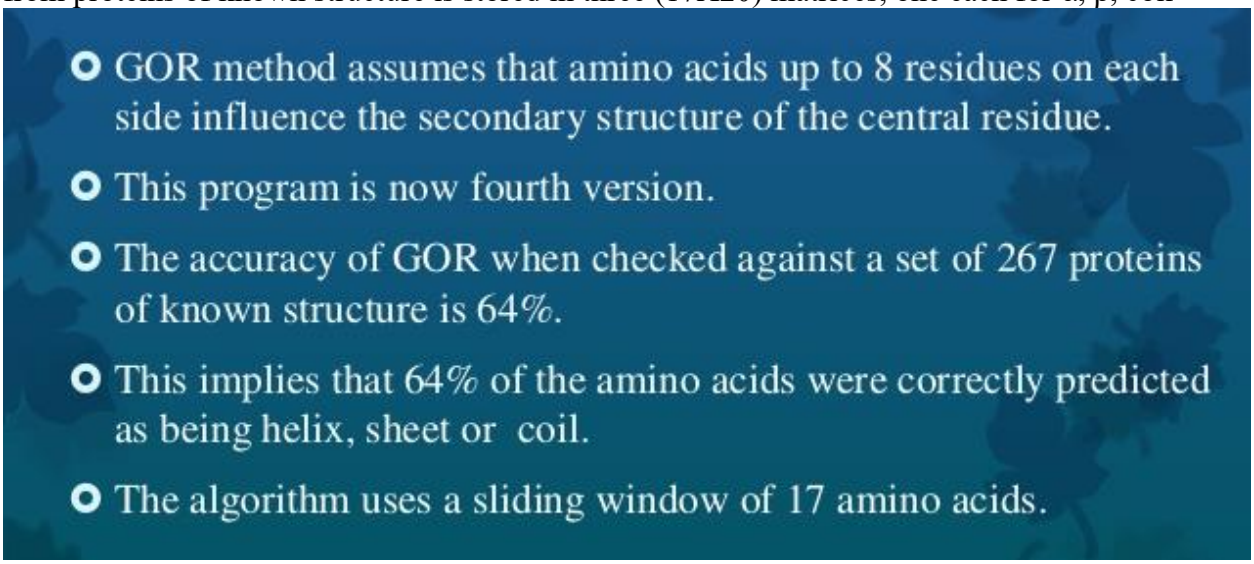


result

- Accuracy: ~50% → ~60%
- helix →alanine,glutamine,leucine,methionine
- Helix breaking →proline and glycine
- Beta sheet →isoleucine,valine,tyrosine
- Beta breaking →proline,aspargine,glutamine
- Turn contains →proline(30%),serine(14%),lysine, aspargine(10%)
- Glycine(19%),aspartic acid (`18%),serine(13%),tyrosine(11%)
- http://www.accelrys.com/product/gcg-wisconsin-package/program-list.html

GOR (Garnier–Osguthorpe–Robson)

GOR Consider window of 17 positions and see how the conformation of the central residuum depends on this residuum and its 18 neighbors (8 in each direction). Ideally one would consider all possible combinations of these neighbors. This is impossible: would require collecting statistics for 20^17 sequences. Instead assume the central residuum depends on its neighbors but the neighbors are independent on each other Implementation :Statistical information derived from proteins of known structure is stored in three (17X20) matrices, one each for α, β, coil



The secondary structure predictions are usually compared with DSSP (Kabsch and Sander, 1983) assignments of secondary structure from crystallographically determined coordinates. Although DSSP defines eight different structural elements, these eight states are commonly translated into three secondary structure states: α-helix, β-sheet and coil. This translation is usually performed in the following manner: (1) α-helix in the three letter code corresponds to H (α-helix), G (3_{10} helix) and I (π-helix) from the DSSP 8-letter code, (2) sheet corresponds to B (bridge—single residue sheet), and E (extended β-strand) in DSSP nomenclature and finally, (3) coil in 3-letter code corresponds to the remaining three DSSP states: T (β-turn), S (bend) and C (coil).

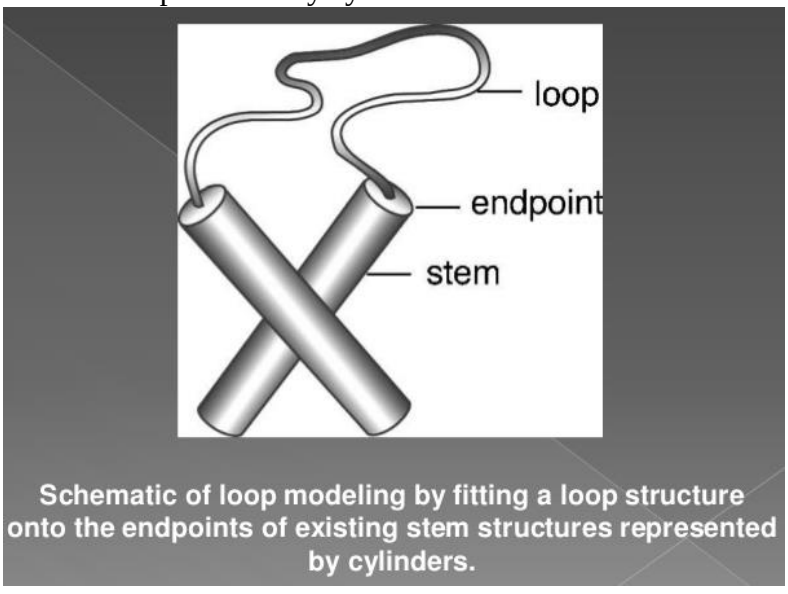
Tertiary structure prediction.

- Protein three-dimensional structures are obtained using two popular experimental techniques, x-ray crystallography and nuclear magnetic resonance (NMR) spectroscopy. There are many important proteins for which the sequence information is available, but their three- dimensional structures remain unknown. Therefore, it is often necessary to obtain approximate protein structures through computer modeling.
- Having a computer-generated three- dimensional model of a protein of interest has many ramifications, assuming it is reasonably correct. It may be of use for the rational design of biochemical experiments, such as site-directed mutagenesis, protein stability, or functional analysis.
- There are three computational approaches to protein three-dimensional structural modeling and prediction.
- They are **homology modeling, threading, and ab initio prediction.**
    1. The first two are knowledge-based methods; they predict protein structures based on knowledge of existing protein structural information in databases.
    2. The ab initio approach is simulation based and predicts structures based on physicochemical principles governing protein folding without the use of structural templates.

Homology modelling

- As the name suggests, homology modeling predicts protein structures based on sequence homology with known structures. It is also known as comparative modeling. The principle behind it is that if two proteins share a high enough sequence similarity, they are likely to have very similar three-dimensional structures. If one of the protein sequences has a known structure, then the structure can be copied to the unknown protein with a high degree of confidence.
- The overall homology modeling procedure consists of six major steps and one additional step.
- **1. Template Selection :-** The template selection involves searching the Protein Data Bank (PDB) for homologous proteins with determined structures. The search can be performed using a heuristic pairwise alignment search program such as BLAST or FASTA.
    1. However, programming based search programmes such as SSEARCH or ScanPS can result in more sensitive search results.
    2. Homology models are classified into 3 areas in terms of their accuracy and reliability.
        1. Midnight Zone: Less than 20% sequence identity. The structure cannot reliably be used as a template.
        2. Twilight Zone: 20% - 40% sequence identity. Sequence identity may imply structural identity.
        3. Safe Zone: 40% or more sequence identity. It is very likely that sequence identity implies structural identity
- Often, multiple homologous sequences may be found in the database. Then the sequence with the highest homology must be used as the template.

- **2. Sequence Alignment** : Once the structure with the highest sequence similarity is identified as a template, the full-length sequences of the template and target proteins need to be realigned using refined alignment algorithms to obtain optimal alignment.
- Incorrect alignment at this stage leads to incorrect designation of homologous residues and therefore to incorrect structural models. Therefore, the best possible multiple alignment algorithms, such as Praline and T-Coffee should be used for this purpose.
- **3. Backbone Model Building** : Once optimal alignment is achieved, the coordinates of the corresponding residues of the template proteins can be simply copied onto the target protein.
- If the two aligned residues are identical, coordinates of the side chain atoms are copied along with the main chain atoms. If the two residues differ, only the backbone atoms can be copied.
- **4. Loop Modelling :** In the sequence alignment for modeling, there are often regions caused by insertions and deletions producing gaps in sequence alignment.
- The gaps cannot be directly modeled, creating "holes" in the model. Closing the gaps requires loop modeling which is a very difficult problem in homology modeling and is also a major source of error. Currently, there are two main techniques used to approach the problem: the database searching method and the ab initio method. The database method involves finding "spare parts" from known protein structures in a database that fit onto the two stem regions of the target protein.
- The stems are defined as the main chain atoms that precede and follow the loop to be modeled. The best loop can be selected based on sequence similarity as well as minimal steric clashes with the neighboring parts of the structure. The conformation of the best matching fragments is then copied onto the anchoring points of the stems. The ab initio method generates many random loops and searches for the one that does not clash with nearby side chains and also has reasonably low energy and φ and ψ angles in the allowable regions in the Ramachandran plot.
- Schematic of loop modeling by fitting a loop structure onto the endpoints of existing stem structures represented by cylinders.



Schematic of loop modeling by fitting a loop structure onto the endpoints of existing stem structures represented by cylinders.

- FREAD is a web server that models loops using the database approach. PETRA is a web server that uses the ab initio method to model loops. CODA is a web server that uses a consensus method based on the prediction results from FREAD and PETRA.
- **5. Side Chain Refinement :** Once main chain atoms are built, the positions of side chains that are not modeled must be determined.
- A side chain can be built by searching every possible conformation at every torsion angle of the side chain to select the one that has the lowest interaction energy with neighboring atoms. Most current side chain prediction programs use the concept of rotamers, which are favored side chain torsion angles extracted from known protein crystal structures. A collection of preferred side chain conformations is a rotamer library in which the rotamers are ranked by their frequency of occurrence.
- In prediction of side chain conformation, only the possible rotamers with the lowest interaction energy with nearby atoms are selected. A specialized side chain modeling program that has reasonably good performance is SCWRL, which is a UNIX program.
- **6. Model Refinement** : In these loop modeling and side chain modeling steps, potential energy calculations are applied to improve the model.

- Modeling often produces unfavorable bond lengths, bond angles, torsion angles and contacts. Therefore, it is important to minimize energy to regularize local bond and angle geometry and to relax close contacts and geometric chain. The goal of energy minimization is to relieve steric collisions and strains without significantly altering the overall structure. However, energy minimization has to be used with caution because excessive energy minimization often moves residues away from their correct positions.
- GROMOS is a UNIX program for molecular dynamic simulation. It is capable of performing energy minimization and thermodynamic simulation of proteins, nucleic acids, and other biological macromolecules. The simulation can be done in vacuum or in solvents. A lightweight version of GROMOS has been incorporated in SwissPDB Viewer.
- **7. Model Evaluation :** The final homology model has to be evaluated to make sure that the structural features of the model are consistent with the physicochemical rules. This involves checking anomalies in φ–ψ angles, bond lengths, close contacts, and so on. If structural irregularities are found, the region is considered to have errors and has to be further refined.
    1. Procheck is a UNIX program that is able to check general physicochemical parameters such as φ–ψ angles, chirality, bond lengths, bond angles, and so on.
    2. WHAT IF is a comprehensive protein analysis server that has many functions, including checking of planarity, collisions with symmetry axes, proline puckering, anomalous bond angles, and bond lengths.
    3. Few other programs for this step are ANOLEA, Verify3D, ERRAT, WHATCHECK, SOV etc.

Threading or Fold recognition

- By definition, threading or structural fold recognition predicts the structural fold of an unknown protein sequence by fitting the sequence into a structural database and selecting the best-fitting fold. The comparison emphasizes matching of secondary structures, which

are most evolutionarily conserved. The algorithms can be classified into two categories, pairwise energy based and profile based.

- **Pairwise Energy Method**: In the pairwise energy based method, a protein sequence is searched for in a structural fold database to find the best matching structural fold using energy-based criteria. The detailed procedure involves aligning the query sequence with each structural fold in a fold library. The alignment is performed essentially at the sequence profile level using dynamic programming or heuristic approaches.
- Local alignment is often adjusted to get lower energy and thus better fitting. The next step is to build a crude model for the target sequence by replacing aligned residues in the template structure with the corresponding residues in the query. The third step is to calculate the energy terms of the raw model, which include pairwise residue interaction energy, solvation energy, and hydrophobic energy.
- Finally, the models are ranked based on the energy terms to find the lowest energy fold that corresponds to the structurally most compatible fold.
- **Profile Method:** In the profile-based method, a profile is constructed for a group of related protein structures. The structural profile is generated by superimposition of the structures to expose corresponding residues.
- Statistical information from these aligned residues is then used to construct a profile. The profile contains scores that describe the propensity of each of the twenty amino acid residues to be at each profile position. To predict the structural fold of an unknown query sequence, the query sequence is first predicted for its secondary structure, solvent accessibility, and polarity. The predicted information is then used for comparison with propensity profiles of known structural folds to find the fold that best represents the predicted profile.
- Threading and fold recognition assess the compatibility of an amino acid sequence with a known structure in a fold library. If the protein fold to be predicted does not exist in the fold library, the method will fail. 3D-PSSM, GenThreader, Fugue are few web based programmes used for threading.

***Ab initio* Prediction**

- When no suitable structure templates can be found, Ab Initio methods can be used to predict the protein structure from the sequence information only. As the name suggests, the ab initio prediction method attempts to produce all- atom protein models based on sequence information alone without the aid of known protein structures.
- Protein folding is modeled based on global free-energy minimization. Since the protein folding problem has not yet been solved, the ab initio prediction methods are still experimental and can be quite unreliable. One of the top ab initio prediction methods is called Rosetta, which was found to be able to successfully predict 61% of structures (80 of 131) within 6.0 Å RMSD (Bonneau et al., 2002).
- The basic idea of Rosetta is: • To narrow the conformation searching space with local structure predictions & • Model the structures of proteins by assembling the local structures of segments
- The Rosetta method is based on assumptions: • Short sequence segments have strong local structural biases & • Multiplicity of these local biases are highly sequence dependent

- 1st step of Rosetta: • Fragment libraries for each 3- & 9-residue segment of the target protein are extracted from the protein structure database using a sequence profile-profile comparison method
- 2nd step of Rosetta: • Tertiary structures are generated using a MC search of the possible combinations of likely local structures, & • Minimizing a scoring function that accounts for nonlocal interactions such as: ⎫ compactness, ⎫ hydrophobic burial, ⎫ specific pair interactions (disulfides & electrostatics), & ⎫ strand pairing

Domain identification using PROSITE.

A **protein domain** is a region of the protein's polypeptide chain that is self-stabilizing and that folds independently from the rest. Each domain forms a compact folded three-dimensional structure. Many proteins consist of several domains. One domain may appear in a variety of different proteins. Molecular evolution uses domains as building blocks and these may be recombined in different arrangements to create proteins with different functions. In general, domains vary in length from between about 50 amino acids up to 250 amino acids in length.[1] The shortest domains, such as zinc fingers, are stabilized by metal ions or disulfide bridges. Domains often form functional units, such as the calcium-binding EF hand domain of calmodulin. Because they are independently stable, domains can be "swapped" by genetic engineering between one protein and another to make chimeric proteins.

The concept of the **domain** was first proposed in 1973 by Wetlaufer after X-ray crystallographic studies of hen lysozyme and papain and by limited proteolysis studies of immunoglobulins. Wetlaufer defined domains as stable units of protein structure that could fold autonomously. In the past domains have been described as units of:

- compact structure
- function and evolution
- folding.

Each definition is valid and will often overlap, i.e. a compact structural domain that is found amongst diverse proteins is likely to fold independently within its structural environment. Nature often brings several domains together to form multidomain and multifunctional proteins with a vast number of possibilities. In a multidomain protein, each domain may fulfill its own function independently, or in a concerted manner with its neighbours. Domains can either serve as modules for building up large assemblies such as virus particles or muscle fibres, or can provide specific catalytic or binding sites as found in enzymes or regulatory proteins.

Several motifs pack together to form compact, local, semi-independent units called domains.[6] The overall 3D structure of the polypeptide chain is referred to as the protein's tertiary structure. Domains are the fundamental units of tertiary structure, each domain containing an individual hydrophobic core built from secondary structural units connected by loop regions. The packing of the polypeptide is usually much tighter in the interior than the exterior of the domain producing a solid-like core and a fluid-like surface. Core residues are often conserved in a protein family, whereas the residues in loops are less conserved, unless they are involved in the protein's function. Protein tertiary structure can be divided into four main classes based on the secondary structural content of the domain.

- All-α domains have a domain core built exclusively from α-helices. This class is dominated by small folds, many of which form a simple bundle with helices running up and down.
- All-β domains have a core composed of antiparallel β-sheets, usually two sheets packed against each other. Various patterns can be identified in the arrangement of the strands, often giving rise to the identification of recurring motifs, for example the Greek key motif.[26]
- α+β domains are a mixture of all-α and all-β motifs. Classification of proteins into this class is difficult because of overlaps to the other three classes and therefore is not used in the CATH domain database.[15]
- α/β domains are made from a combination of β-α-β motifs that predominantly form a parallel β-sheet surrounded by amphipathic α-helices. The secondary structures are arranged in layers or barrels

Limits on size

Domains have limits on size.[27] The size of individual structural domains varies from 36 residues in E-selectin to 692 residues in lipoxygenase-1,[18] but the majority, 90%, have fewer than 200 residues[28] with an average of approximately 100 residues.[29] Very short domains, less than 40 residues, are often stabilised by metal ions or disulfide bonds. Larger domains, greater than 300 residues, are likely to consist of multiple hydrophobic cores

Multidomain Proteins

The majority of proteins, two-thirds in unicellular organisms and more than 80% in metazoa, are multidomain proteins. However, other studies concluded that 40% of prokaryotic proteins consist of multiple domains while eukaryotes have approximately 65% multi-domain proteins.

Many domains in eukaryotic multidomain proteins can be found as independent proteins in prokaryotes, suggesting that domains in multidomain proteins have once existed as independent proteins. For example, vertebrates have a multi-enzyme polypeptide containing the GAR synthetase, AIR synthetase and GAR transformylase domains (GARs-AIRs-GARt; GAR: glycinamide ribonucleotide synthetase/transferase; AIR: aminoimidazole ribonucleotide synthetase). In insects, the polypeptide appears as GARs-(AIRs)2-GARt, in yeast GARs-AIRs is encoded separately from GARt, and in bacteria each domain is encoded separatel

Interpro

InterPro provides functional analysis of proteins by classifying them into families and predicting domains and important sites. To classify proteins in this way, InterPro uses predictive models, known as signatures, provided by several different databases (referred to as member databases) that make up the InterPro consortium. We combine protein signatures from these member databases into a single searchable resource, capitalising on their individual strengths to produce a powerful integrated database and diagnostic tool

Database	Description	URL
Blocks	Database of protein alignment blocks	http://blocks.fhcrc.org

Database	Description	URL
CDD	Conserved domain database	http://www.ncbi.nlm.nih.gov/Structure/cdd/cdd.shtml
CluSTr	Clusters of SWISS-PROT and TrEMBL proteins	http://www.ebi.ac.uk/clustr/
DOMO	Protein-domain database based on sequence alignments	http://www.infobiogen.fr/services/domo/
InterPro	Integrated documentation resource for protein families, domains and functional sites	http://www.ebi.ac.uk/interpro/
IProClass	Integrated protein classification database	http://pir.georgetown.edu/iproclass/
MetaFam	Database of protein family information	http://metafam.ahc.umn.edu/
Pfam	Collection of multiple sequence alignments and hidden Markov models	http://www.sanger.ac.uk/Software/Pfam/
PIR	Protein Information Resource	http://pir.georgetown.edu/
PIR-ALN	Curated database of protein sequence alignments	http://pir.georgetown.edu/pirwww/dbinfo/piraln.html
PRINTS-S	Compendium of protein fingerprints	http://www.bioinf.man.ac.uk/dbbrowser/PRINTS/
ProClass	Non-redundant protein database organized by family relationships	http://pir.georgetown.edu/gfserver/proclass.html
ProDom	Automatic compilation of homologous domains	http://prodes.toulouse.inra.fr/prodom/doc/prodom.html
PROSITE	Database of patterns and profiles describing protein families and domains	http://www.expasy.ch/prosite/
ProtoMap	Automatic hierarchical classification of SWISS-PROT proteins	http://www.protomap.cs.huji.ac.il/
SBASE	Curated protein domain library based on sequence clustering	http://www3.icgeb.trieste.it/~sbasesrv/
SMART	Simple Modular Architecture Research Tool - a collection of protein families and domains	http://smart.embl-heidelberg.de/
SWISS-PROT and TrEMBL	Protein sequence databases	http://www.ebi.ac.uk/swissprot/ or http://www.expasy.org/sprot/

Database	Description	URL
SYSTERS	Systematic re-searching method for sequence searching and clustering	http://systers.molgen.mpg.de/
TIGRFAMs	Protein families based on hidden Markov models	http://www.tigr.org/TIGRFAMs/