

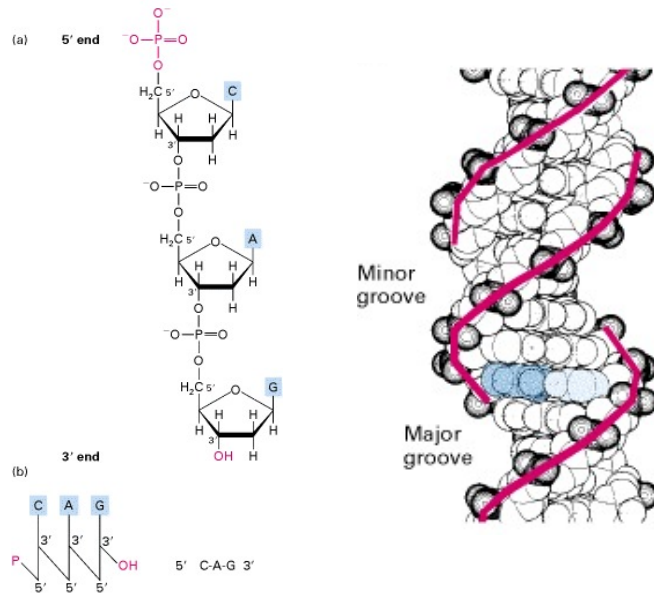
Bioinformatics & (Computational) Molecular Biology

Introduction

Alexander (Sacha) Gultyaev
a.p.goultiaev@liacs.leidenuniv.nl

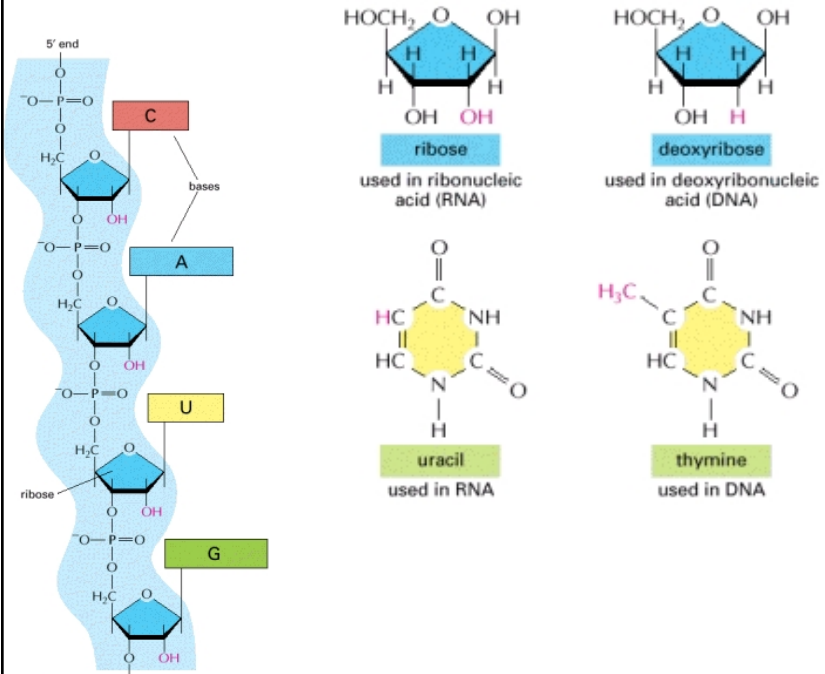
Biopolymer sequences

DNA: double-helical nucleic acid.
Monomers: nucleotides C, A, T, G.



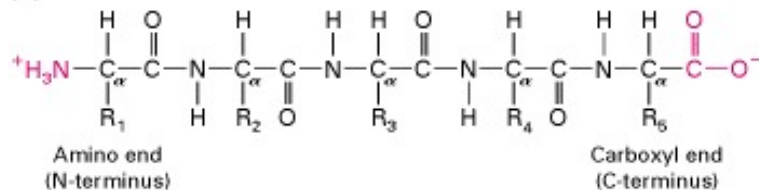
from "Molecular Cell Biology", Lodish et al. (2000)

RNA: (single-stranded) nucleic acid.
Monomers: nucleotides C, A, U, G.




from "Molecular Biology of the Cell", Alberts et al. (2002)

Proteins: polypeptide chains. Monomers: amino acids (20 types).



from "Molecular Cell Biology", Lodish et al. (2000)

Biopolymer sequences

 replication (DNA polymerase)

Watson-Crick complementarity:
AT and **GC** pairs in double helix

DNA

5' – ATGGCGCAGGG...–3'
3' –TACCGCGTCCC...–5'



transcription (RNA polymerase)

RNA

5' – AUGGCGCAGGG...–3'



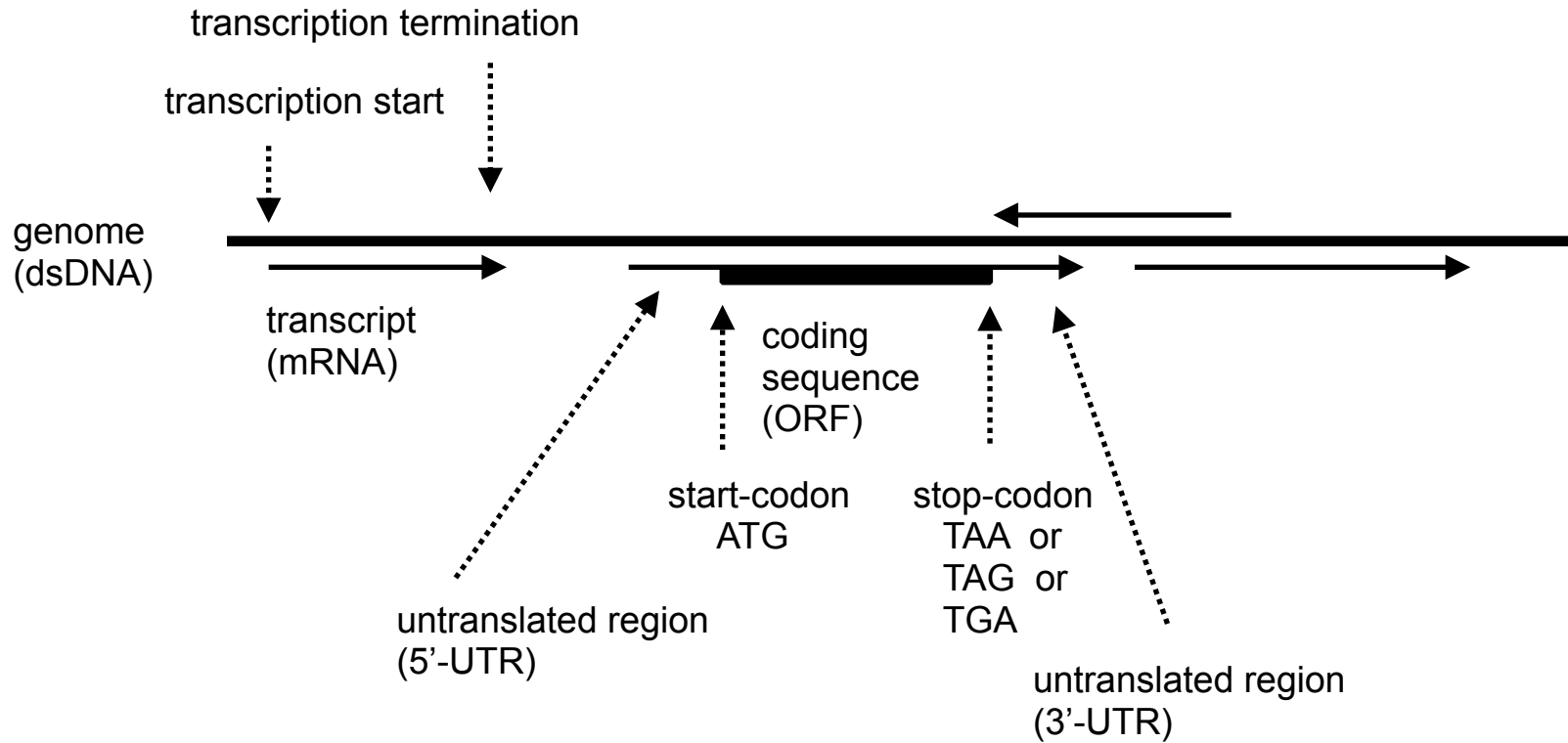
translation (ribosome)

protein

MetAlaGlnGly...

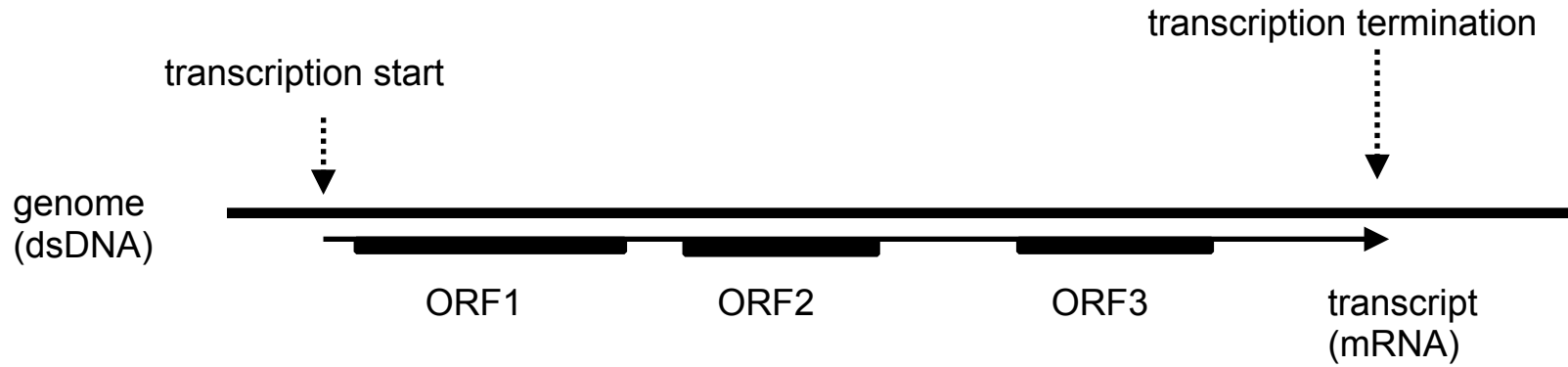
Genetic code is redundant:
 $4 \times 4 \times 4 = 64$ nucleotide triplets
encode for 20 amino acids. Only
two amino acids (Met and Trp) are
encoded by a single codon. Other
amino acids can be encoded by 2,
4 or 6 codons. Usually the last
“wobble” position of a triplet
determines “silent” substitutions.
ATG (Met) is start codon (usually).
TAA, TAG, TGA - stop codons.

Basic signals in gene expression



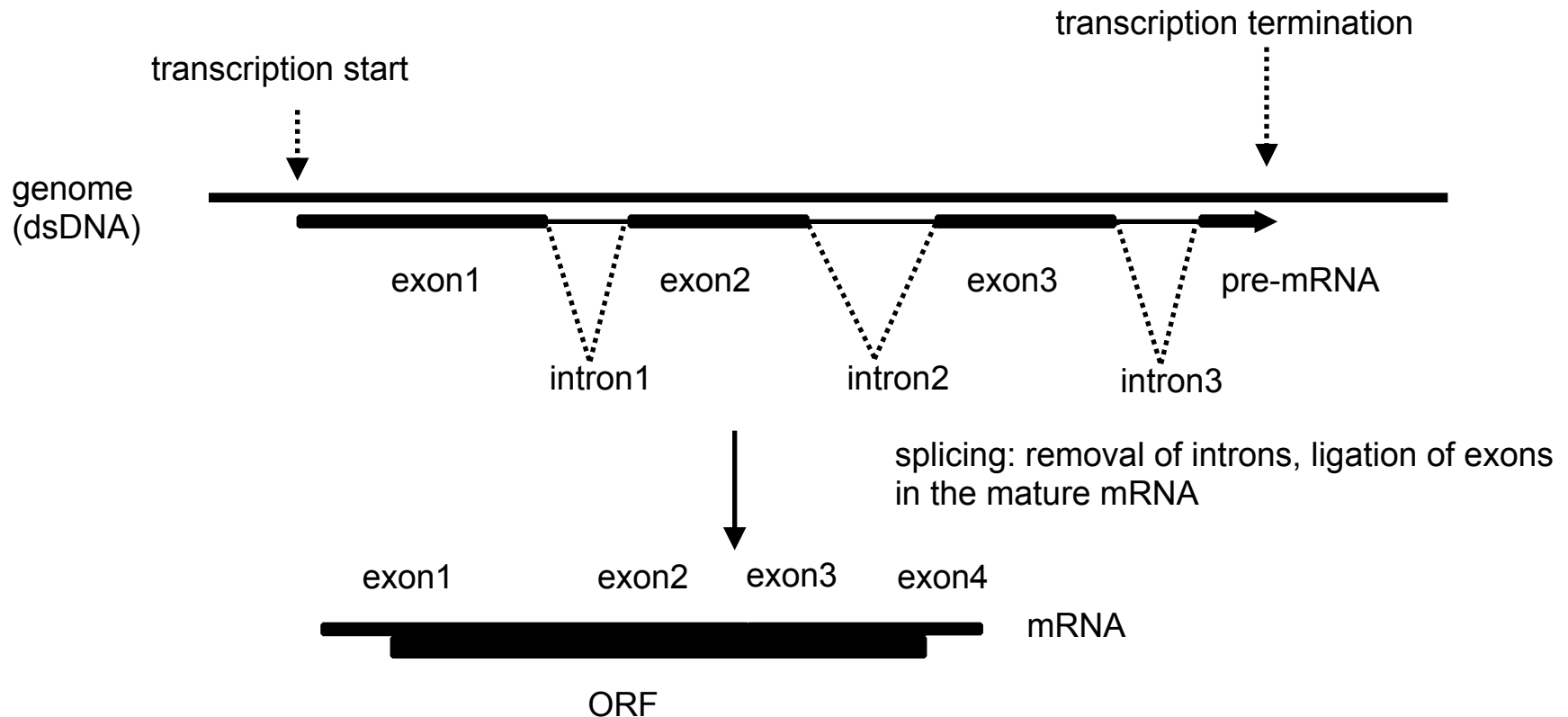
Basic signals in gene expression

Prokaryotes (bacteria): polycistronic mRNA



Basic signals in gene expression

Eukaryotes: precursor-mRNA (pre-mRNA) processing (splicing)



Alternative splicing, such as exon skipping or intron retention, leads to diverse isoforms of mRNAs and proteins encoded by the same gene. Due to frameshifts the sequences of proteins could be different.

NCBI database resources / Entrez retrieval system

Safari File Edit View History Bookmarks Window Help

National Center for Biotechnology Information

www.ncbi.nlm.nih.gov

PseudoBase Mathews Lab...Astructure UNAFold | m...albany.edu TBI - Vienn...A Package 2 Clustal Ome...< EMBL-EBI Apple iCloud Facebook Twitter Wikipedia Yahoo! News Popular

NCBI Resources How To Sign in to NCBI

NCBI National Center for Biotechnology Information

All Databases Search

NCBI Home

Resource List (A-Z)

- All Resources
- Chemicals & Bioassays
- Data & Software
- DNA & RNA
- Domains & Structures
- Genes & Expression
- Genetics & Medicine
- Genomes & Maps
- Homology
- Literature
- Proteins
- Sequence Analysis
- Taxonomy
- Training & Tutorials
- Variation

Welcome to NCBI

The National Center for Biotechnology Information advances science and health by providing access to biomedical and genomic information.


[About the NCBI](#) | [Mission](#) | [Organization](#) | [Research](#) | [NCBI News](#)

Get Started

- [Tools](#): Analyze data using NCBI software
- [Downloads](#): Get NCBI data or software
- [How Tos](#): Learn how to accomplish specific tasks at NCBI
- [Submissions](#): Submit data to GenBank or other NCBI databases

Genomic Structural Variation

dbVar archives large scale genomic variation data and associates defined variants with phenotypic information.



1 2 3 4 5 6 7 8

Popular Resources

- PubMed
- Bookshelf
- PubMed Central
- PubMed Health
- BLAST
- Nucleotide
- Genome
- SNP
- Gene
- Protein
- PubChem

NCBI Announcements

Milestone: NCBI annotates 150th eukaryotic genome

Aug 28, 2014

NCBI has now completed the genome annotation for 150 different organisms.

The new NCBI Genomes FTP site is here!

Nucleotide sequence databases

Initially three main databases: GenBank (USA),
EMBL (Europe),
DDBJ (Japan).

Later the three databases became parts of
the **International Nucleotide Sequence Database Collaboration**.

The three organizations exchange data on a daily basis.

Each record is assigned a unique identifier, **Accession number**, that is shared by three databases.

A single flat file format of database entries is used.

Datafields of sequence database entries:

LOCUS NM_000518 626 bp mRNA linear PRI 24-MAY-2014
DEFINITION Homo sapiens hemoglobin, beta (HBB), mRNA.
ACCESSION NM_000518
VERSION NM_000518.4 GI:28302128
KEYWORDS RefSeq.
SOURCE Homo sapiens (human)
ORGANISM [Homo sapiens](#)
Eukaryota; Metazoa; Chordata; Craniata; Vertebrata; Euteleostomi;
Mammalia; Eutheria; Euarchontoglires; Primates; Haplorrhini;
Catarrhini; Hominidae; Homo.
REFERENCE 1 (bases 1 to 626)
AUTHORS Mei Y, Yin N, Jin X, He J and Yin Z.
TITLE The regulatory role of the adrenergic agonists phenylephrine and
isoproterenol on fetal hemoglobin expression and erythroid
differentiation
JOURNAL Endocrinology 154 (12), 4640-4649 (2013)
PUBMED [24080366](#)

... etc.

FEATURES Location/Qualifiers
source 1..626
/organism="Homo sapiens"
/mol_type="mRNA"
/db_xref="taxon:[9606](#)"
/chromosome="11"
/map="11p15.5"
gene 1..626
/gene="HBB"
/gene_synonym="beta-globin; CD113t-C"
/note="hemoglobin, beta"
/db_xref="GeneID:[3043](#)"
/db_xref="HGNC:[HGNC:4827](#)"
/db_xref="HPRD:[00786](#)"
/db_xref="MIM:[141900](#)"
exon 1..142
/gene="HBB"
/gene_synonym="beta-globin; CD113t-C"
/inference="alignment:Splign:1.39.8"
CDS 51..494
/gene="HBB"
/gene_synonym="beta-globin; CD113t-C"
/note="beta globin chain; hemoglobin beta chain"
/codon_start=1
/product="hemoglobin subunit beta"
/protein_id="[NP_000509.1](#)"
/db_xref="GI:4504349"
/db_xref="CCDS:[CCDS7753.1](#)"
/db_xref="GeneID:[3043](#)"
/db_xref="HGNC:[HGNC:4827](#)"
/db_xref="HPRD:[00786](#)"
/db_xref="MIM:[141900](#)"
/translation="MVHLTPEEKSAVTALWGKVNVDEVGGEALGRLLVVYPWTQRFFESFGDLSTPDVAMGNPKVKAHGKKVLGAFSDGLAHLNLIKGTATLSEHLCDKLHVDPE
NFRLLGNVLVLCVLAHFGKEFTPPVQAAYQKVVAGVANALAHKYH"

... etc.

ORIGIN
1 acatttgctt ctgacacaac tgtgttcact agcaacctca aacagacacc atggtgcac
61 tgactcctga ggagaagtct gccgttactg ccctgtggg caaggtgaac gtggatgaag

Header

Features

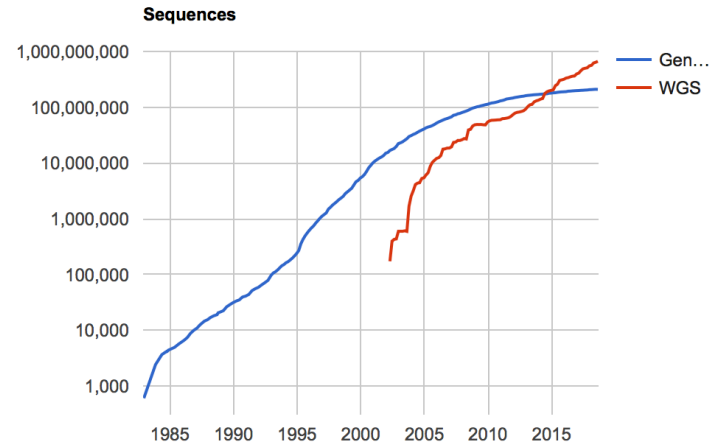
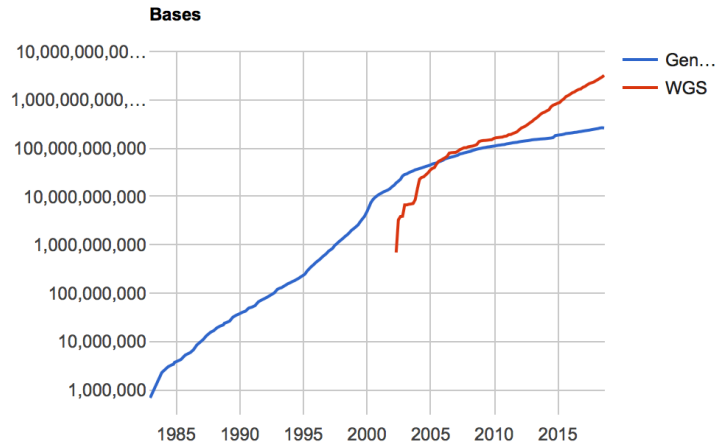
Sequence

Examples of features annotated:

gene	1..7898 /gene="SRSF7"
mRNA	join(1..266,1301..1481,1790..1966,2842..2916,3338..3448, 4748..4801,6308..7898) /gene="SRSF7"
CDS	join(239..266,1301..1481,1790..1966,2842..2916,3338..3448, 4748..4801,6308..6362) /gene="SRSF7"
CDS	complement(46224..48638) /locus_tag="KPHS_00400" /codon_start=1 /transl_table= 11 /product="formate dehydrogenase-O alpha subunit" /protein_id=" YP_005224340.1 " /db_xref="GeneID: 11845018 "
exon	697..832
regulatory	7153..7158 /regulatory_class="polyA_signal_sequence"

Statistics of nucleotide sequence databases

GenBank and WGS Statistics



		GenBank		WGS	
Release	Date	Bases	Sequences	Bases	Sequences
3	Dec 1982	680338	606		
14	Nov 1983	2274029	2427		
20	May 1984	3002088	3665		
● ● ● ● ● ● ● ● ● ● ● ● ●					
225	Apr 2018	260189141631	208452303	2784740996536	621379029
226	Jun 2018	263957884539	209775348	2944617324086	639804105
227	Aug 2018	260806936411	208831050	3204855013281	665309765

[<https://www.ncbi.nlm.nih.gov/genbank/statistics/>, accessed 15.09.2018]

Databases of amino acid sequences

Historically: the first databases.

1965: The Atlas of Protein Sequences and Structures.

Nowadays amino acid sequences are predominantly determined by translation of massively sequenced nucleic acids.

Thus a database of amino acid sequences is secondary or curated database.

(In contrast to e.g. primary GenBank with records obtained from submitters.)

ENTREZ protein database: a collection of entries from several databases such as SWISS-PROT (one of the oldest and popular databases) and translations of nucleotide sequences in GenBank.

UniProtKB (uniprot.org): Knowledgebase, contains both amino acid sequences and functional annotation.

- includes SWISS-PROT (manually annotated and reviewed) and
- TrEMBL (suggested coding regions in the nucleotide database entries, automatically annotated and not reviewed).

Note that a number of coding regions (ORFs) in the nucleotide database entries may remain unannotated.

The Reference Sequence (RefSeq) Database

Non-redundant, richly annotated records of nucleotide and amino acid sequences.

RefSeq entries are similar to those of GenBank, but they have some distinct features, in particular, specific Accession prefixes.

Accession prefix	Molecule type	Comment
AC_	Genomic	Complete genomic molecule, usually alternate assembly
NC_	Genomic	Complete genomic molecule, usually reference assembly
NG_	Genomic	Incomplete genomic region
NT_	Genomic	Contig or scaffold, clone-based or WGS ^a
NW_	Genomic	Contig or scaffold, primarily WGS ^a
NS_	Genomic	Environmental sequence
NZ_ ^b	Genomic	Unfinished WGS
NM_	mRNA	
NR_	RNA	
XM_ ^c	mRNA	Predicted model
XR_ ^c	RNA	Predicted model
AP_	Protein	Annotated on AC_ alternate assembly
NP_	Protein	Associated with an NM_ or NC_ accession
YP_ ^c	Protein	
XP_ ^c	Protein	Predicted model, associated with an XM_ accession
ZP_ ^c	Protein	Predicted model, annotated on NZ_ genomic records

From: Chapter 18, The Reference Sequence (RefSeq) Database



The NCBI Handbook [Internet].
McEntyre J, Ostell J, editors.
Bethesda (MD): National Center for Biotechnology Information (US); 2002-.

Entrez Gene database

Gene-centered database. Integrates info from multiple databases. Typically an entry follows the annotation of RefSeq entries. The records have multiple links to other databases.

A fragment of an entry of the Gene database:

Transcript accessions

Intron

Exon

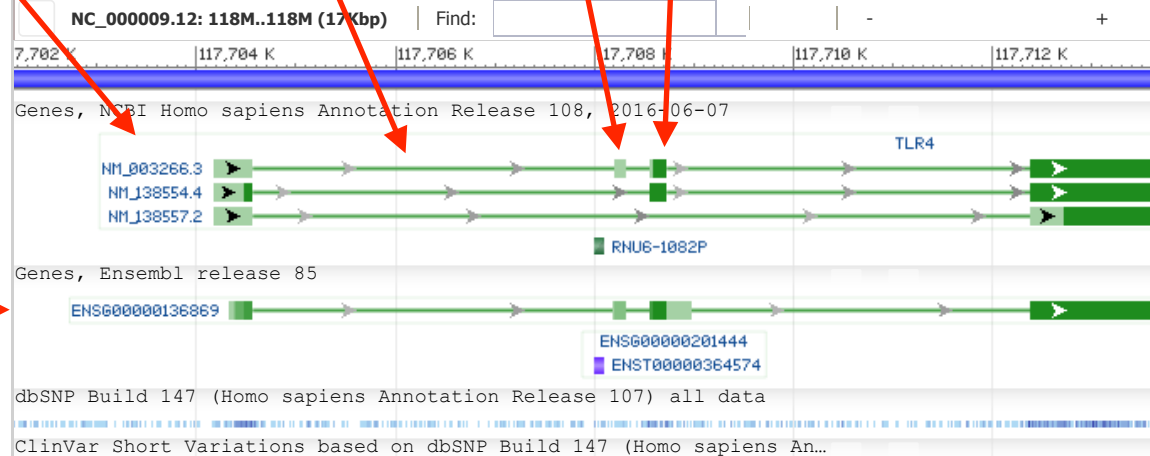
Coding region

Genomic regions, transcripts, and products

Go to [reference sequence details](#)

Genomic Sequence: NC_000009.12 Chromosome 9 Reference GRCh38.p7 Primary Assembly

Go to nucleotide: [Graphics](#) [FASTA](#) [GenBank](#)



Genome browser transcript data

Database of known mutations (dbSNP)

Sequence alignment

- Sequence alignment is used for identification of *homology* and/or *similarity* of sequences.

Homology (evolutionary history) is not equivalent to similarity (e.g. % identity).

- However, identification of sequence similarity helps to reveal the homology.
- Similarity of 1D sequences (primary structures) can be seen in sequence monomers of two sequences mapped against each other:

```
Sequence A  GCTTA----GCTATTGGCTTCTCTAAT--CACCAAGGGATATGCATACAAAAACATTCT
              |  | | |      | | | | | | | | | | | | | | | | | | | | | | | | | | | |
Sequence B  GATTATTTAGCTATTGGCTTCTTTAATAATAACCATTGATATG-----GAAAAATTTCT
```

Sequences can be aligned in many different ways

```
Sequence A  GC----TTAGCTATTGGCTTCTCTAATC--ACCAAGGGATATGCATACAAAAACATTCT
              |  | | | | | | | | | | | | | | | | | | | | | | | | | | | |
Sequence B  GATTATTTAGCTATTGGCTTCTTTAATAATAACCATTGATA-----TGGAAAAATTTCT
```

↑

Insertion A->B or deletion B->A (alignment “gap”)

↑↑

Substitutions

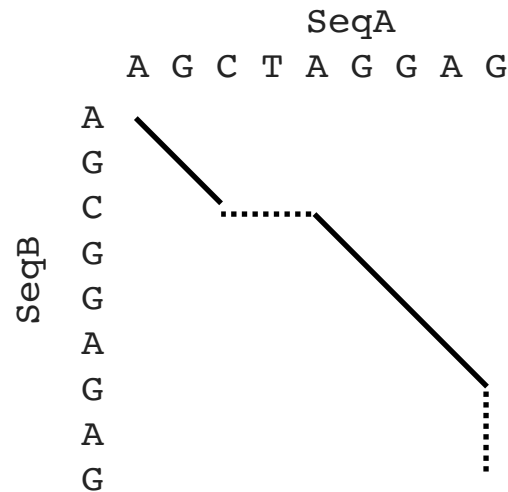
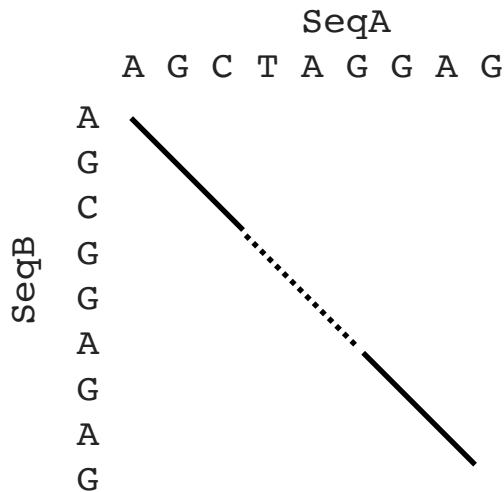
Alignment algorithms attempt to identify most likely alignment, trying to follow the molecular mechanisms of sequence evolution: substitutions, deletions, insertions.

Searching for optimal alignment

Alternative alignments can be viewed as alternative paths in 2D sequence space.

SeqA AGCTAGGAG
 ||| |||
SeqB AGCGGAGAG

SeqA AGCTAGGAG--
 ||| |||
SeqB AGC--GGAGAG



Most likely alignment should contain “as-large-as-possible” number of most likely events (conservation of monomers -> matched positions) and “as-small-as-possible” number of less likely events (substitutions, gaps). It is possible to assign some **scores** to all alignment elements according to their probabilities in biologically relevant model.

An alignment can be **scored**, and so can be the corresponding path.

Searching for optimal alignment

Two separate issues in finding the optimal alignment:

1. Scoring system.
2. Algorithm to find the alignment with the best score
(optimal alignment = optimal path).

Scoring may be relatively simple, for instance, the default parameters of the BLASTN program for alignment of nucleotide sequences:

(+2) for match;

(-3) for mismatch;

(-5) for the first gap nucleotide and (-2) for each of the nucleotides in gap extension
(negative **“penalties”**)

Scoring system should be derived from observed substitution frequencies in homologous sequences

During DNA replication, transitions ($A \leftrightarrow G$, $C \leftrightarrow T$) are more frequent as compared to transversions ($A \leftrightarrow C$, $A \leftrightarrow T$, $G \leftrightarrow T$, $G \leftrightarrow C$). This can be taken into account by a **substitution matrix**, e.g. as shown below:

	A	C	G	T
A	91	-114	-31	-123
C	-114	100	-125	-31
G	-31	-125	100	-114
T	-123	-31	-114	91

Identities have positive scores, substitutions - negative penalties.

Diagonal elements should not be equal, reflecting differences in occurrence of various nucleotides.

Amino acid substitution matrices

Amino acid substitution matrices take into account so-called “conservative” substitutions between residues with similar properties (e.g. Arg \leftrightarrow Lys).

The scores for a 20×20 matrix can be derived from frequencies observed in the datasets of related proteins.

These frequencies should be computed as log-odds: logarithms of ratios of the frequencies to the background ones that are determined by chance.

$$s(a,b) = \frac{1}{\lambda} \log \frac{p_{ab}}{f_a f_b}$$

Here a and b are two residue types, p_{ab} is observed frequency, f_a and f_b are occurrences of a and b , respectively, in all proteins. λ is a scaling factor to make the scores convenient integers.

Gap penalties

Insertions or deletions (indels) are less frequent than point substitutions, and are therefore penalized in alignments by negative scores.

There is no reliable theoretical basis for gap statistics.

Usually a linear function for gap penalty $S(\text{gap})$ for a gap of n monomers:

$$S(\text{gap}) = G + n \times L$$

Parameters G (gap opening penalty) and L (gap length or extension) are chosen empirically. The optimal choice is dependent on substitution matrices and expected similarity of aligned sequences.

For instance, $G = 10$ and $L = 1$ can be used in combination with BLOSUM62.

In alignments of nucleotide sequences the following parameters are chosen as default in BLASTN program for sequence database similarity search:

$G = 3$ and $L = 2$ in combination with match = 2 and mismatch = -3 .

Searching for optimal alignment

Given a scoring system, the score of any alignment can be computed.

The problem is, however, to find the **optimal** alignment with the best score.

Even for alignment of two sequences of 300 monomers, about 10^{179} alignments are possible...

Various types of alignment:

Pairwise alignment : global (full length) or local (finding the best aligned regions).

Sequence database similarity search: given a query sequence, find the best aligned sequences in the database.

Multiple sequence alignment: alignment of some number (>2) of sequences.

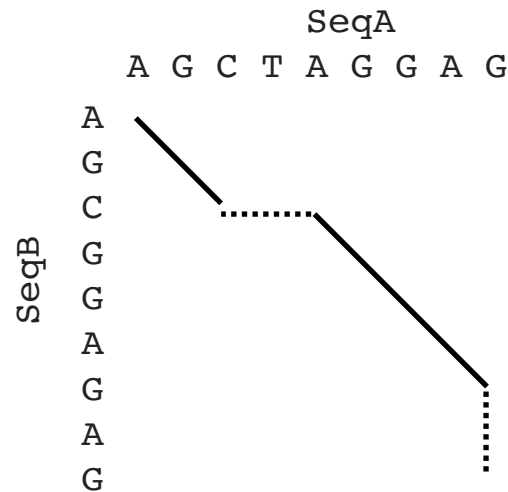
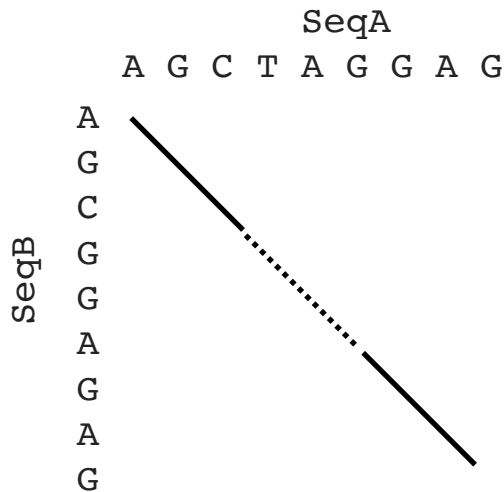
Different algorithms are designed for each of these problems.

Searching for optimal alignment

Alternative alignments can be viewed as alternative paths in 2D sequence space.

SeqA AGCTAGGAG
 | | | | | |
SeqB AGCGGAGAG

SeqA AGCTAGGAG--
 | | | | | |
SeqB AGC--GGAGAG



An alignment can be **scored**, and so can be the corresponding path.

↘ Diagonal move: match or mismatch.

→
↓ Vertical or horizontal move: gap.

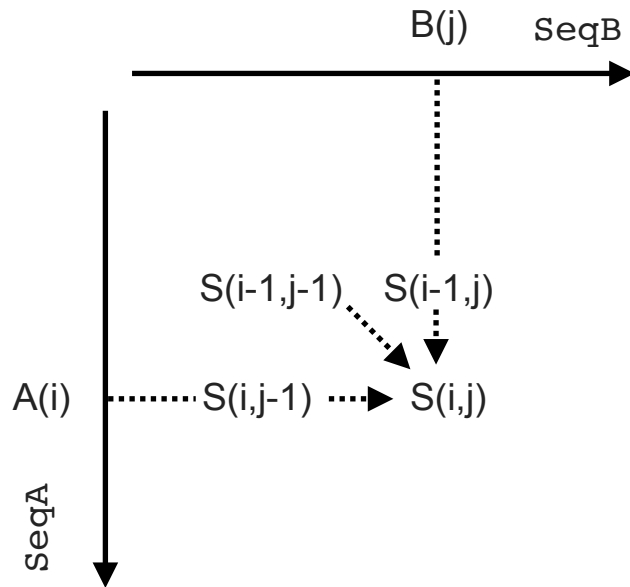
Searching for optimal alignment by a dynamic programming algorithm

Recursive calculation of the optimal alignment score $S(i,j)$:

$$S(i,j) = \max \begin{cases} S(i,j) + M_{\text{subst}}[A(i), B(j)], \\ S(i,j-1) + G, \\ S(i-1,j) + G. \end{cases}$$

M_{subst} - substitution matrix of the scoring system;

G - gap penalty.



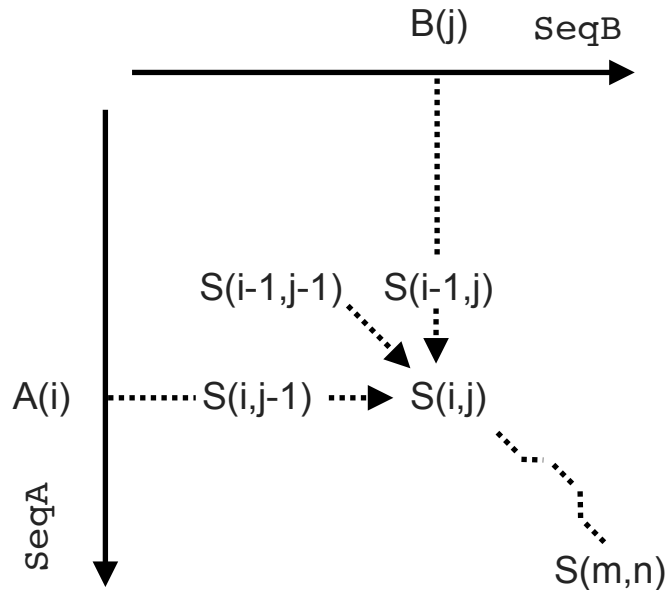
Searching for optimal alignment by a dynamic programming algorithm

Recursive calculation of the optimal alignment score $S(i,j)$:

$$S(i,j) = \max \begin{cases} S(i,j) + M_{\text{subst}}[A(i),B(j)], \\ S(i,j-1) + G, \\ S(i-1,j) + G. \end{cases}$$

M_{subst} - substitution matrix of the scoring system;

G - gap penalty.



The recursive formula allows the calculation of dynamic programming matrix starting from smaller subalignments.

All elements of the matrix $S(i,j)$ correspond to the optimal scores of partial alignments.

The score $S(m,n)$, where m and n are two sequence lengths, is the optimal global alignment score.

The optimal alignment can be retrieved by backtracking of all moves that have led to $S(m,n)$.

A toy example (here match is +5; mismatch is - 2 and insertion/deletion is - 6):

Dynamic programming matrix:

		j → (sequence y)								
		0	1	2	3	4	5	6	7	8 = N
			T	G	C	T	C	G	T	A
i ↓ (sequence x)	0	0	-6	-12	-18	-24	-30	-36	-42	-48
	1 T	-6	5	-1	-7	-13	-19	-25	-31	-37
	2 T	-12	-1	3	-3	-2	-8	-14	-20	-26
	3 C	-18	-7	-3	8	2	3	-3	-9	-15
	4 A	-24	-13	-9	2	6	0	1	-5	-4
	5 T	-30	-19	-15	-4	7	4	-2	6	0
M = 6	A	-36	-25	-21	-10	1	5	2	0	11

Optimum alignment scores 11:

T	-	-	T	C	A	T	A
T	G	C	T	C	G	T	A
+5	-6	-6	+5	+5	-2	+5	+5

Needleman - Wunsch algorithm:
optimal **global** alignment using dynamic programming.

Waterman - Smith algorithm:
optimal **local** alignment using dynamic programming.

Optimal local alignment is defined as the alignment of subregions of two sequences with the maximum score.

NB. It does not mean that such subregions are aligned separately, they are identified by the dynamic programming matrix constructed for full-length sequences.

The programs exploiting these algorithms are available via the ENTREZ system and in the EMBL-EBI tools.

Needleman - Wunsch algorithm:
optimal **global** alignment using dynamic programming.

Waterman - Smith algorithm:
optimal **local** alignment using dynamic programming.

Optimal local alignment is defined as the alignment of subregions of two sequences with the maximum score.

NB. It does not mean that such subregions are aligned separately, they are identified by the dynamic programming matrix constructed for full-length sequences.

The programs exploiting these algorithms are available via the ENTREZ system and in the EMBL-EBI tools.

“**Glocal**” alignment: global on one of the sequences, local on the other.

(Reasonable in some cases, e.g. when one of the sequences is expected to be homologous to a domain within the other).

Can be computed e.g. with zero gap penalties at the ends of one of the sequences.

Estimates of alignment significance:

Global alignments: no accurate statistical theory. The reliability of an alignment can be estimated using multiple alignments of permutations of aligned sequences. If the score of alignment of interest is significantly higher than the average score obtained from pairs of sequences of the same lengths and compositions (permutations), it is judged to be significant rather than determined by chance alone.

Say, the optimal global alignment of seqA and seqB has the score S_{AB} .

seqA = {GAGCTAA...}

seqB = {GCAAGCC...}

↓ *permutations of sequences, like e.g.
perm{12345} → {32451} or {43152} etc.*

Alignment score [perm(seqA), perm(seqB)] → S_1

↓ *repeat e.g. 100 times*

Average (S_1, S_2, \dots, S_{100}) = S_{avg}

If S_{AB} is significantly higher than S_{avg} ,
the alignment is significant.

Estimates of alignment significance:

Local alignments: expected number (E-value) of ungapped local alignments with score at least S in the alignment of sequences with sufficiently large lengths m and n :

$$E = K m n \exp (- \lambda S),$$

where K and λ depend on scoring system and monomer frequencies.

No general theory for gapped alignments. Statistics can be estimated using quasi-random sequences.

Sequence database similarity search

- Input: sequence query
- Output: list of similar sequences (“hits”) found in the database

- Sequence database similarity search implies pairwise alignments of the query to all entries in the database.
- A straightforward dynamic programming algorithm is not efficient in this case (slow).
- A faster search can be realized using search for “words”: stretches of similar oligomers in two sequences (the query and a subject sequence from the database).

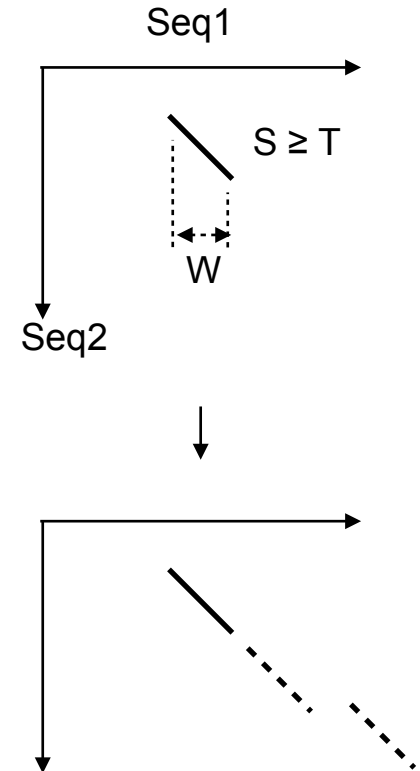
BLAST: Basic Local Alignment Search Tool

BLAST is the most popular program for sequence database similarity search.

First publication: Altschul et al. (1990).

Main strategy:

- Searching for “words” in a subject sequence from the database satisfying a criterion of a word size at least W and a score (S) at least T compared to a word in the query.
- If a word is found, BLAST algorithm attempts to extend it and improve the score S .
- The algorithm is designed for local alignments: if further extension does not improve S , the alignment region between the query and the subject sequence (“sequence hit”) with the maximal S is returned to the user.
- The result of BLAST is a list of hits, ordered according to their significance (E-values).



BLAST: Basic Local Alignment Search Tool

Say, searching with a query: ...FDRIGDGETKL**VTP**VPT...

“w-mers”: words that score at least T when compared to some word (e.g. **VTP**) in the query.

With W=3; T=11 and BLOSUM62 matrix, w-mer scores calculated for VTP:

VTP 16	MTP 13	CTP 11	VSP 12	VVP 11
ITP 15	ATP 12	FTP 11	VAP 11	
LTP 13	TTP 12	YTP 11	VNP 11	

Subject ...VDQHGAPPEQR**ITP**RQQ...

contains ITP (S=15) => the algorithm proceeds with the extension phase
(e.g. alignment by dynamic programming)

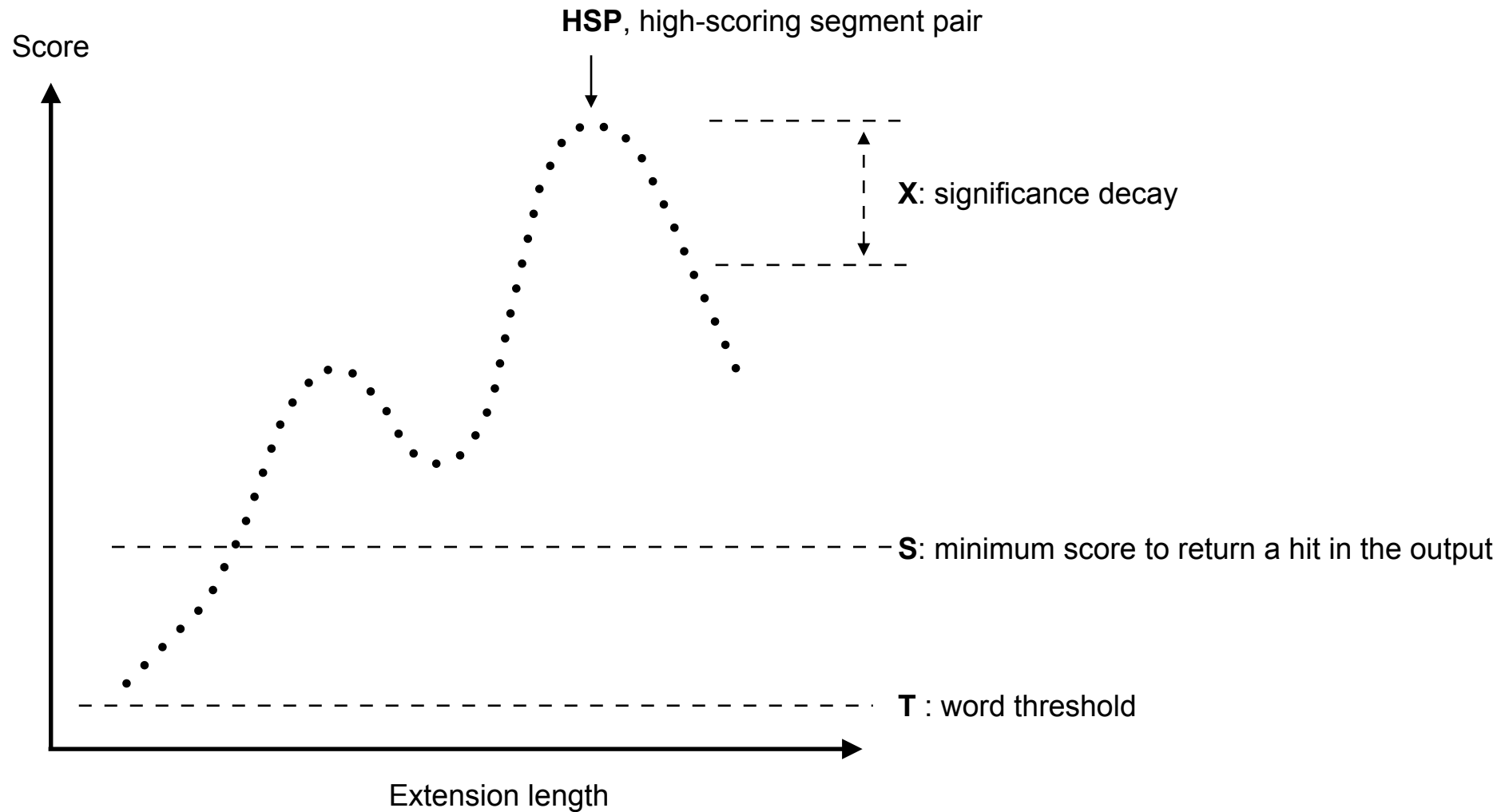
Query ...FDRIGDGETKL**VTP**VPT...
Sbjct ...VDQHGAPPEQR**ITP**RQQ...



Score improved ?

Word extension search in the original BLAST algorithm

(Altschul et al., 1990)



The statistics of pairwise alignments

Expected number (E-value) of ungapped HSPs with score at least S in the alignment of sequences with sufficiently large lengths m and n :

$$E = K m n \exp (- \lambda S),$$

where K and λ depend on scoring system and monomer frequencies.

Normalized raw score

$$S' = (\lambda S - \ln K) / \ln 2$$

is a “bit score” characterizing HSP significance : $E = m n 2^{-S'}$
(not dependent on scoring system).

For gapped local alignments the statistics can be determined from large-scale comparisons of quasi-random sequences.

Gapped BLAST

(Altschul et al., 1997)

Two-hit approach: initial search for two non-overlapping hits of score at least T , within a distance A of one another on a diagonal in sequence space.

Advantage: faster search without losing significant sequence similarities.

Two-hit approach: initial search for two non-overlapping hits of score at least T , within a distance A of one another on a diagonal in sequence space:



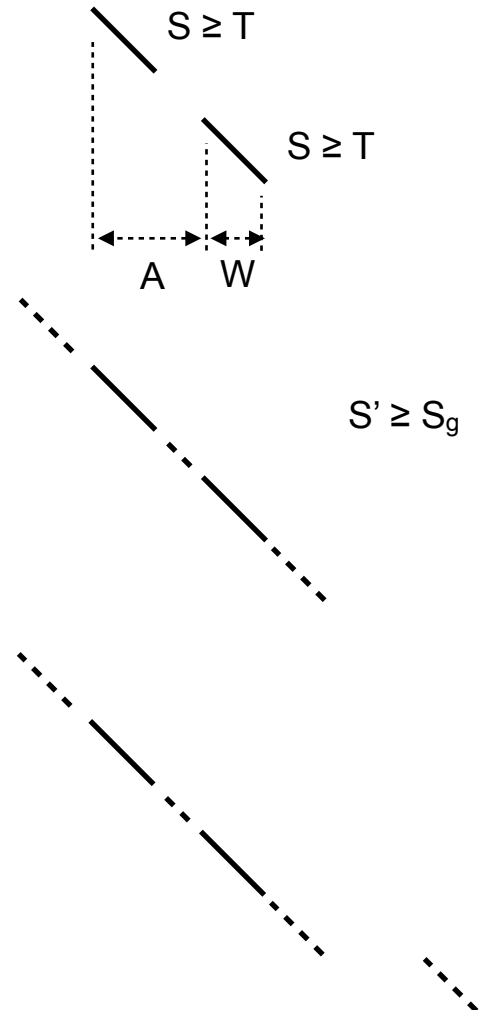
Ungapped extension:



If ungapped extension is better than some threshold S_g .
E.g. chosen so that not more than one gapped extension is
invoked per 50 database sequences, corresponding to
 $S_g = 22$ bits:



Gapped extension is triggered.



BLAST webpage


BLAST® » **blastn suite**


Standard Nucleotide BLAST

blastnblastpblastxtblastntblastx


BLASTN programs search nucleotide databases using a nucleotide query. [more...](#)

Enter Query Sequence


Enter accession number(s), gi(s), or FASTA sequence(s)  [Clear](#)


Query subrange 
From
To

Or, upload file

Choose File no file selected 

Job Title



Enter a descriptive title for your BLAST search 

☐ Align two or more sequences 



Choose Search Set

Database

☐ Human genomic + transcript ☐ Mouse genomic + transcript ☒ Others (nr etc.):

Nucleotide collection (nr/nt)  

Organism

Optional
 Enter organism name or id—completions will be suggested ☐ Exclude 
Enter organism common name, binomial, or tax id. Only 20 top taxa will be shown 



Exclude

Optional
☐ Models (XM/XP) ☐ Uncultured/environmental sample sequences

Limit to


Optional
☐ Sequences from type material

Entrez Query

Optional
  [Create custom database](#)
Enter an Entrez query to limit search 

Program Selection

Optimize for

☒ Highly similar sequences (megablast)
☐ More dissimilar sequences (discontiguous megablast)
☐ Somewhat similar sequences (blastn)
Choose a BLAST algorithm 

BLAST

Search **database Nucleotide collection (nr/nt)** using **Megablast (Optimize for highly similar sequences)**
☐ Show results in a new window

Multiple sequence alignment

Multiple: $N > 2$

N=5 :

Hfq_Bsubtilis	--MKPINIQDQFLNQIRKENTYVTVFLLNGFQLRGQVKGFDFNFTVLLESEGKQQLIYKHA
Hfq_Lpneumophila	-MSKNHLLQDPFLNELRKEKVPVSVFLVNGIKLHGIIDSFDQYVVMLKN-SITQMVYKHA
Hfq_Ecoli	-MAKGQSLQDPFLNALRRERVPSIYLVNGIKLQGQIESFDQFVILLKN-TVSQMVYKHA
Hfq_Ngonorrhoeae	MTAKGQMLQDPFLNALRKEHVPVSIYLVNGIKLQGQVESFDQYVVLLRNTSVTQMVYKHA
Hfq_Neuropaea	MGVKGQLLQDPFLNILRKERIPVSIYLVNGIKLQGQIDSFDQYVVLLKN-SVTQMVYKHA

Hfq_Bsubtilis	ISTFAPQKNVQLELE-----
Hfq_Lpneumophila	ISTVVPSRMVKIPAEESSGEEGTAD-----
Hfq_Ecoli	ISTVVPSRPVSHHSNAGGGTSSNYHHGSSAQNTSAQQDSEETE
Hfq_Ngonorrhoeae	ISTIVPARSVNLQHENKPQAAPASTL----VQVETVQQPAE---
Hfq_Neuropaea	ISTIVPAKAISIPIPADTQTEQDEP-----

- *An accurate alignment of multiple sequences by direct application of dynamic programming is not feasible (computationally demanding, could be applied only for small datasets of relatively short sequences).*
- *Various MSA strategies are used for faster algorithms. One of the most straightforward ones: progressive multiple sequence alignment.*

Progressive multiple sequence alignment

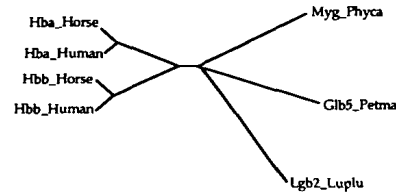
Progressive MSA: an algorithm starts from aligning the closely related sequences, with following iterations consisting of aligning the previously built alignments. At every iteration, a pairwise alignment of two clusters of sequences is carried out.

ClustalW
(Thompson et al., 1994):

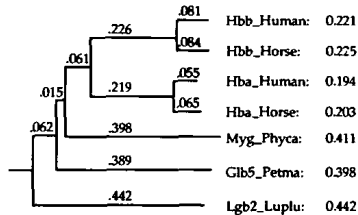
Pairwise alignment:
Calculate distance matrix

Hbb_Human	1	-					
Hbb_Horse	2	.17	-				
Hba_Human	3	.59	.60	-			
Hba_Horse	4	.59	.59	.13	-		
Myg_Phyca	5	.77	.77	.75	.75	-	
Glb5_Petma	6	.81	.82	.73	.74	.80	-
Lgb2_Luplu	7	.87	.86	.86	.88	.93	.90
		1	2	3	4	5	6

Unrooted Neighbor-Joining tree



Rooted NJ tree (guide tree)
and sequence weights



Progressive alignment:
Align following
the guide tree

```

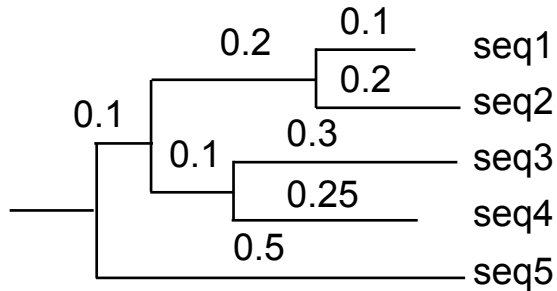
-----VRLSPERKAVTALNGKV--PDEVGGEALGRLLVVETGCRFESFGDLST
-----VQLSPERKAAVLALNDKVN--EEVVGGEALGRLLVVETGCRFESFGDLST
-----VLSFADKTHVKAAMGKVDAAGHTGAALEHGMFLSPFTKHTFPHFDLS--
-----VLSFADKTHVKAAMGKVDAAGHTGAALEHGMFLSPFTKHTFPHFDLS--
-----VLSFGEWQLVLEWAKVDAAGHTGAALEHGMFLSPFTKHTFPHFDLS--
PIVDTGSAVPLSAEKTKIRSAMAPVSTSTSGVDILVGFTHSTPAQHPFPFKGLTT
-----GALTESQAALVKSWEETPAAPKHTHRTFVLAETAPAKLPSPLKGTSE

PDAVMGNSKVKAHQKKVLAQVSDGTAHLD----NLGTFATLSEHCDLKVDEENFRL
PDAVMGNSKVKAHQKKVLESPGEGVHLD----NLGTFATLSEHCDLKVDEENFRL
---HGAQVKAHQKKVADALTAHVHVD----DLGALSHLSDLHAEKLRVDPVNFKL
---HGAQVKAHQKKVQDALTAHVHLD----DLGALSHLSDLHAEKLRVDPVNFKL
EAMGKASDLKQGVTVLTALGAILCKG----HRAELKPLAQSHATCKIKIKYLEF
ADQLKGSADVQMAERIKAVHDAVSHDDT---EKEMKLDLSGTHAHSFQVCPQYFKV
VF--QNDPELOARAGKVKVLYTEAAQLQVTVVVTDATLKNLGSVHYSG--VADHFFV

LGMVLVCVLAHSPGKEFTFPVQAIVQKVAGVANALAKTH-----
LGMVLVCVLAHSPGKEFTFPVQAIVQKVAGVANALAKTH-----
LSHCLLVTLAHSPPAEFTPAVHABLDKFLASVSTVLTSKYR-----
LSHCLLVTLAHSPPAEFTPAVHABLDKFLASVSTVLTSKYR-----
ISEAIIHVLESHPGDPGADAQQAQMKALELFRDIAAKYKELGTQG
LAAVIADTVAG-----DAPFERLMSMICILLRSAY
VKEALIKTKIKVWGAKNSELSNHTLAIDELAIVIGKMDAA--
  
```

Using sequence weights in progressive multiple alignment

• E.g. ClustalW algorithm:



rooted tree with
branch lengths

$$W(1) = 0.1 + 0.2/2 + 0.1/4 = 0.225$$

$$W(2) = 0.2 + 0.2/2 + 0.1/4 = 0.325$$

$$W(3) = 0.3 + 0.1/2 + 0.1/4 = 0.375$$

$$W(4) = 0.25 + 0.1/2 + 0.1/4 = 0.325$$

$$W(5) = 0.5/1 = 0.5$$

the contributions of branch lengths to the weights are
divided by cluster sizes

• Sequences are aligned according to the tree order. At each step, dynamic programming is used for pairwise alignment of (clusters of) sequences. The substitution scores are calculated as weighted averages of scores for substitutions between clusters.

```
...VLLESEGGKQQL... seq1
...VMLKN-SITQM... seq2
.....(i).....
```

For instance, alignment of two clusters
{seq1,seq2} vs {seq3,seq4}

```

. .
V I
L L
L L
R K(j)
N N
. .
s s
e e
q q
4 3
```

$$S(i, j) = S(E, K) \times W(1) \times W(3) +$$

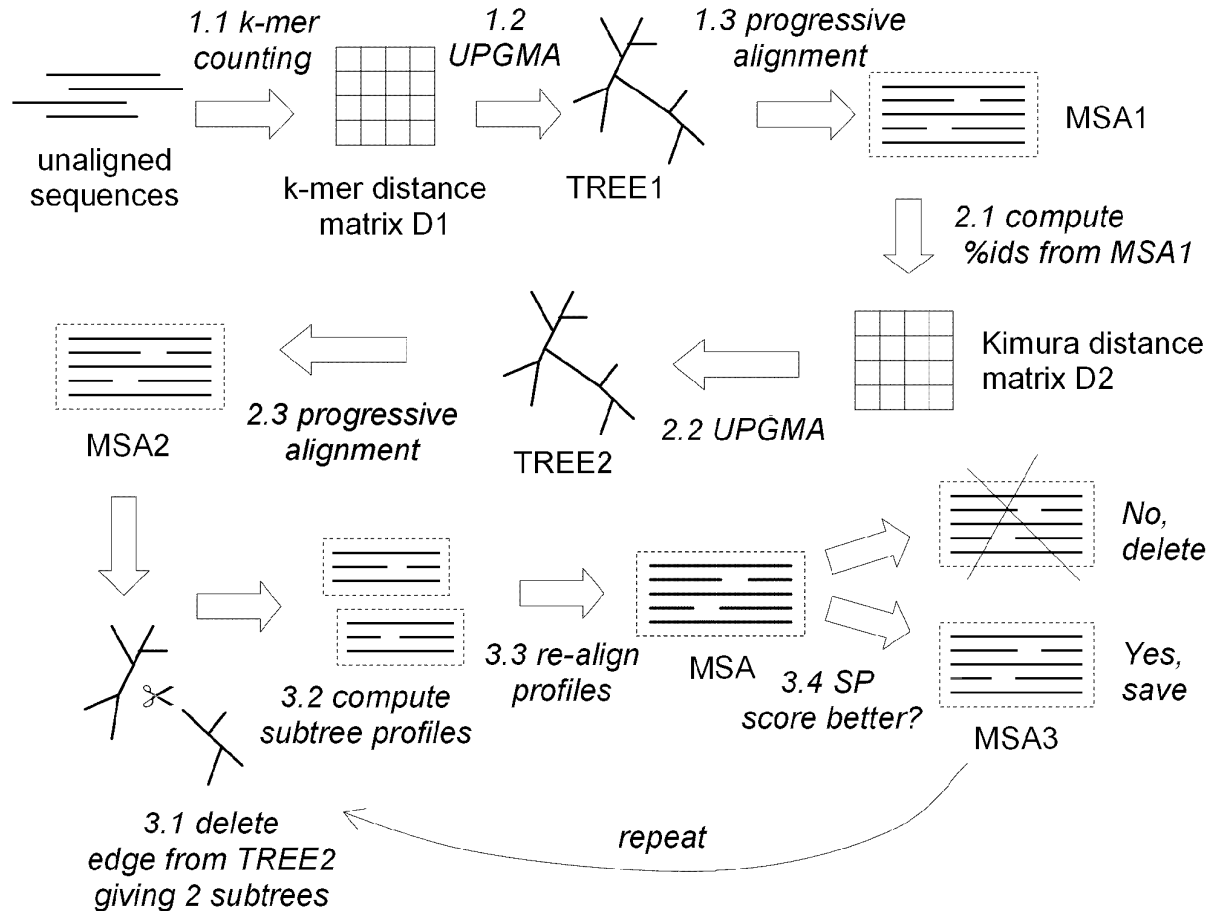
$$+ S(E, R) \times W(1) \times W(4) +$$

$$+ S(K, K) \times W(2) \times W(3) +$$

$$+ S(K, R) \times W(2) \times W(4)$$

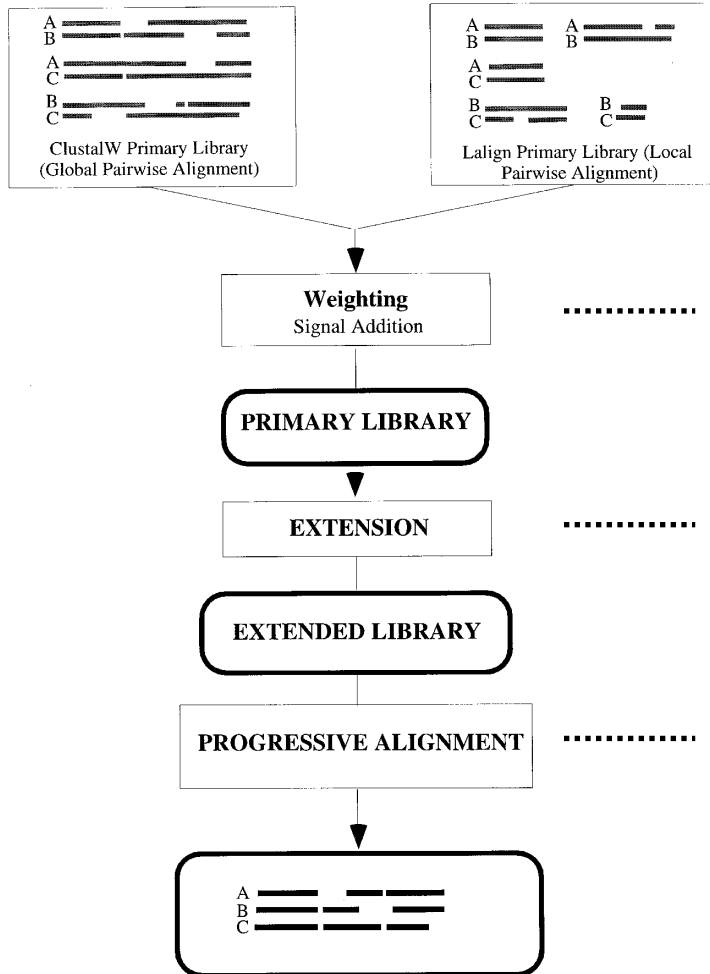
Multiple sequence alignment: various strategies

e.g. MUSCLE
(R.C. Edgar, 2004):



Multiple sequence alignment: various strategies

e.g. T-Coffee
(Notredame et al., 2004):



Lalign local: 10 top-scoring non-intersecting local alignments

- Monomer $x(A)$ aligned to $y(B)$: constraint.
- $W(\text{constraint}) = \% \text{ similarity in the alignment.}$
- $W(x,y) = W(x,y, \text{global}) + W(x,y, \text{local}).$
- $W = 0$ if x and y are not aligned.

- If $x(A)$ is aligned to $z(C)$ and $y(B)$ is aligned to $z(C)$ as well:
 x and y are aligned through sequence C , thus additional constraint weight:
 $W(x,y) = W(x,y) + \min [W(x,z) + W(y,z)].$

- Progressive alignment according to the NJ tree from pairwise alignments.
- Dynamic programming is carried out with account of weights.
- No gap penalties (indirectly they are already taken into account).

Alignment-free sequence comparisons (*k*-mer analysis)

Similar sequences have similar “word” compositions.

Words: L-tuples or k-mers.

Comparisons of these compositions can be done faster than those based on alignments.

Important for large datasets of sequences.

Say, sequences $X = \text{AAACTGGT}\dots \rightarrow$ 6-mers: $\text{AAACTG}, \text{AACTGG}, \text{ACTGGT}, \dots$
 $Y = \text{AGAACTGG}\dots \rightarrow$ $\text{AGAACT}, \text{GAACTG}, \text{AACTGG}, \dots$
 $Z = \text{AAATTGGT}\dots \rightarrow$ $\text{AAATTG}, \text{AATTGG}, \text{ATTGGT}, \dots$

=> In this region X and Y share one 6-mer, Z is different.

For a given k , a sequence X can be converted into vector

$$C(X,k) = (c_{X,k,1}, c_{X,k,2}, \dots c_{X,k,N})$$

($c_{X,k,i}$ is word count for the i -th k -mer and N is number of all possible k -mers.)

Different metrics for a distance $d(X,Y)$ are possible, for instance:

$$d(X,Y) = \sum_{i=1}^N (c_{X,k,i} - c_{Y,k,i})^2 \quad (\text{Euclidean distance})$$

or fractional common k -mer count F :

$$F(X,Y) = \sum_{i=1}^N \min(c_{X,k,i}, c_{Y,k,i}) / [\min(\text{length}X, \text{length}Y) - k + 1]$$

(Here an upper limit of homologous i -th k -mers in two sequences is normalised by the maximum number of homologous k -mers. This value decreases with increasing evolutionary distance).