Introductory Bioinformatics For Users: Course Notes

Author: Bela Tiwari

November 1, 2007

Contents

Chapter 1

Introduction

What is bioinformatics? There are many different answers to this. One basic definition is that it is the use of computational methods to analyse biological data.

What this course will cover: This course will introduce some of the many resources available for analysing sequence data. We focus on tools available via the web, but during the lectures, some benefits and drawbacks of this approach will be discussed and alternatives suggested. The course starts by reviewing some of the major data resources publicly available and how they can be accessed. Sequence alignments and scoring systems employed will be reviewed, and activities will showcase a few pieces of commonly available software. Methods to search sequence databases and the statistics reported will be covered, with a special emphasis on the family of blast-based search tools. During the course, advice will be given on topics such as data management and automation with the hopes that this will help you establish efficient routes through your bioinformatics analyses in the future.

What this course will not cover: This course does not give you a firm theoretical grounding for the bioinformatics methods discussed. The underpinning theory is important for understanding the results you obtain. While some background information is given here, it is highly recommended that you read further on any of the programs or methods you choose to use to analyse data for your own research.

In addition, this course focusses purely on basic sequence-based analysis and does not cover topics such as phylogenetics, protein structure, microarrays, proteomics or metabolomics. In addition, important topics such as sequence profiles and profile alignments are only touched on briefly.

By the end of this course, you will

- have accessed some of the many data resources available at the NCBI and EBI,
- understand some of the basic principles behind aligning sequences,

- understand some key points about different sequence alignment programs,
- have experience running some web-based bioinformatics programs,
- understand the information returned by some sequence database searching programs,
- appreciate some of the practical approaches available for automating bioinformatics.

Chapter 2

Sequence and sequence-related databases

There are many freely available data resources. A large number are hosted by large national and international institutions such as the American center, the National Center for Biotechnology Information (NCBI) and the European Bioinformatics Centre (EBI).

There are also many specialist centres and research groups offering data resources. Some of these are available for searching via their own web pages and in some cases.

Many sequence databases can also be downloaded in parts or in their entirety to use locally, or can be accessed at the remote location using web pages, web services and sometimes, specialised software on your own machine.

A word of warning. Many sequences in the database have been annotated by transferring annotation from the most similar sequence in the database already annotated. That sequence in turn may have been annotated the same way. Many sequences known to be quite similar may, in fact, have different function when investigated experimentally rather than purely *in silico*. The lesson is to use evidence codes[1] when they are available to help you judge how much confidence to put in a particular annotation.

2.1 Nucleotide repositories

2.1.1 EMBL, GenBank and DDBJ

There are three main nucleotide repositories: **EMBL** (European), **GenBank** (American), and **DDBF** (Japanese). These repositories hold annotated nucleotide sequences.

[1] For example, look at the PE (protein existence) line in Swissprot. There are 5 evidence codes: **1**: evidence at protein level, **2**: evidence at transcript level, **3**: inferred from homology, **4**: predicted, **5**: uncertain

They have an agreement to share all their data, so which one you choose *shouldn't* matter[2].

Characteristics of entries in the primary nucleotide repositories

- The large nucleotide databases are not hand-curated: the quality of the information is largely dependent on the people submitting the sequence.
- Records can be updated by the original submitter, or by a third party if the submitter granted them permission and notified the relevant institute (not common).
- There are redudant entries in these databases.
- Entries can contradict one another.
- Predicted or known proteins coded for by the sequence are linked to via their accession number in the Uniprot knowledgebase.
- Information from any species, including sequences of unknown origin, can be deposited in the database.

2.1.2 TPA

A relatively new section in the nucleotide databases is the Third Party Annotation (TPA) section. It consists of sequences from the primary collection, but with new annotation that has been published in a peer-reviewed scientific journal. TPA includes two types of records: experimental (supported by wet-lab evidence) and inferential (where the annotation has been inferred by sequence similarity, not shown by direct experimentation).

2.1.3 Anatomy of an EMBL entry

The initial two letters of an EMBL entry indicate what the information in that line contains. An example EMBL entry is given below. This will be discussed further during the class. GenBank entries are very similar to EMBL entries, but their precise formats are somewhat different.

```
ID   EE572316; SV 1; linear; mRNA; EST; INV; 164 BP.
XX
AC   EE572316;
XX
DT   10-SEP-2007 (Rel. 93, Created)
DT   05-OCT-2007 (Rel. 93, Last updated, Version 2)
XX
DE   fUKS10nFBr13 Ugandan isolate UKS10 Pf var gene cDNA Plasmodium falciparum
DE   cDNA clone UKS10nFBr13 similar to Plasmodium falciparum erythrocyte
DE   membrane protein 1, var, mRNA sequence.
```

[2] Due to some issues related to what sections particular types of sequence are stored in and what unique identifiers certain software uses by default, the choice of database can occassionally make a difference. In terms of overall content, you can choose between any of the three.

```
XX
KW   EST.
XX
OS   Plasmodium falciparum (malaria parasite P. falciparum)
OC   Eukaryota; Alveolata; Apicomplexa; Aconoidasida; Haemosporida; Plasmodium;
OC   Plasmodium (Laverania).
XX
RN   [1]
RP   1-164
RX   PUBMED; 17895392.
RA   Normark J., Nilsson D., Ribacke U., Winter G., Moll K., Wheelock C.E.,
RA   Bayarugaba J., Kironde F., Egwang T.G., Chen Q., Andersson B., Wahlgren M.;
RT   "PfEMP1-DBL1{alpha} amino acid motifs in severe disease states of
RT   Plasmodium falciparum malaria";
RL   Proc. Natl. Acad. Sci. U.S.A. 104(40):15835-15840(2007).
XX
CC   Contact: Nilsson D
CC   Department of Cell and Molecular Biology
CC   Karolinska institutet
CC   Berzeliusv 35, SE-17177, Stockholm, Sweden
CC   Tel: 46 8 52483991
CC   Email: daniel.nilsson@ki.se
CC   PCR PRimers
CC   FORWARD: nDBLf
CC   BACKWARD: alpha-BR
CC   Seq primer: M13 forward primer
CC   High quality sequence stop: 164.
XX
FH   Key             Location/Qualifiers
FH
FT   source          1..164
FT                   /organism="Plasmodium falciparum"
FT                   /lab_host="Homo sapiens"
FT                   /isolate="UKS10"
FT                   /mol_type="mRNA"
FT                   /dev_stage="Trophozoite"
FT                   /clone_lib="Ugandan isolate UKS10 Pf var gene cDNA"
FT                   /clone="UKS10nFBr13"
FT                   /tissue_type="Blood"
FT                   /note="Vector: pcrII-TOPO; Blood samples were drawn from
FT                   Ugandan children with mild or severe malaria. Isolate UKS10
FT                   from Kampala, Uganda originated from a patient sufferring
FT                   from severe malaria. Erythrocytes were separated from other
FT                   blood constituents and cultured until parasites matured
FT                   into trophozoites. Trophozoites were enriched, and total
FT                   RNA extracted. Reverse transcription with random hexamer
FT                   oligonucleotides and DNAse treatment gave single stranded
FT                   cDNA. Three different pairs of primers (alpha-AF/alpha-BR,
FT                   nDBLf/nDBLr, nDBLf/alpha-BR) were used to PCR amplify the
FT                   var gene/PfEMP1 DBL1alpha domain from cDNA. A set of 48
FT                   clones were obtained from each reaction and sequenced once
FT                   from each direction."
FT                   /db_xref="taxon:5833"
XX
SQ   Sequence 164 BP; 63 A; 25 C; 32 G; 44 T; 0 other;
     tggcagccaa atatgagggg gacttaataa aaacacgtta tacaccatat caacaaatat        60
     atggtgattc tgcttcccaa ttatgtactg tattagcacg aagtttcgca gatataggcg       120
     atattataag aggaaaagat ctgtatctcg gtgaataaaa aaaa                        164
//
```

2.1.4 RefSeq

RefSeq (Reference Sequence) is a curated collection of DNA, RNA, and protein sequences built by the NCBI. RefSeq provides one example of each natural biological molecule for organisms with enough available information. (In January, 2007, this amounted to data for approximately 4000 different organisms.) For each model organism, RefSeq aims to provide separate and linked records for the genomic DNA, the gene transcripts, and the proteins arising from those transcripts.

To produce RefSeq records, NCBI culls the best available information on each molecule and updates the records as more information emerges. You could think of GenBank as the primary literature of sequences, while RefSeq is more like the review literature of sequences.

RefSeq entries can be easily distinguished from GenBank entries; they have an accession prefix with an underscore, and a notation in the comment field that indicates the RefSeq status[3].

Characteristics of entries in RefSeq

- NCBI staff hand-curate entries using data sources including GenBank.
- Records are revised by NCBI staff as additional information becomes available.
- Single entries for each sequence for an organism.
- Predicted or known proteins coded for by the sequence are linked to via their accession number in the Uniprot knowledgebase. Transcripts are also linked.
- Only information from major model organisms are included in RefSeq.

The software we will be using today is hosted by the European Bioinformatics Institute (EBI), which maintains the EMBL database, and the National Center for Biotechnology Information (NCBI), which is based in the USA and hosts GenBank.

[3] An example of a RefSeq accession id is XP_001271868, while an example GenBank or EMBL accession is DS027054.

2.2 Peptide repositories

2.2.1 Uniprot-knowledgebase

The main peptide database in use is called **Uniprot-knowledgebase**. It is a results of a merger of the efforts of three databases that had earlier been held and curated separately: The Swiss Institute of Bioinformatics and the EBI's **SwissProt** and **TrEMBL** databases, and Georgetown Universitys **PIR-PSD** database.

The history of Uniprot is still apparent in the divisions within it: SwissProt and TrEMBL.

Uniprot-SwissProt is the "gold" standard of public databases; it consists of manually-annotated records. SwissProt records are cross-referenced to more than 50 different databases.

Uniprot-TrEMBL consists of predicted coding sequences (CDS regions) as annotated in the primary nucleotide databases. (There is some additional merging and adjustment before inclusion in TrEMBL.) TrEMBL and SwissProt are non-redundant. Peptides from TrEMBL that are shown to exist and are then annotated are promoted into SwissProt.

Swissprot contains high quality information and is much smaller than TrEMBL; in general SwissProt is a good place to start your searches, whether in searching the metadata or the sequence data itself.

All protein sequences encoded by the same gene are merged into a single UniProt-SwissProt entry. Differences among sequencing reports are analysed and fully described in the feature table (e.g. alternative splicing events, polymorphisms or conflicts).

This course concentrates on accessing sequences via portals with access to many databases such as SRS and Entrez. However, the home sites for many databases can offer intuitive and comphrensive interfaces for data exploration. For example, you might like to visit the new Uniprot website to investigate particular proteins. The interface makes it particularly easy to see the evidence for the annotation included in an entry. The cross reference section is also very intuitive.

2.2.2 Anatomy of a Uniprot entry

```
ID   LDH_PLAFA               Reviewed;          42 AA.
AC   P13774;
DT   01-JAN-1990, integrated into UniProtKB/Swiss-Prot.
DT   01-JAN-1990, sequence version 1.
DT   24-JUL-2007, entry version 35.
DE   L-lactate dehydrogenase (EC 1.1.1.27) (LDH-P) (Fragment).
OS   Plasmodium falciparum.
OC   Eukaryota; Alveolata; Apicomplexa; Aconoidasida; Haemosporida;
OC   Plasmodium; Plasmodium (Laverania).
OX   NCBI_TaxID=5833;
RN   [1]
RP   NUCLEOTIDE SEQUENCE [GENOMIC DNA].
RX   MEDLINE=85240418; PubMed=3892292; DOI=10.1016/0166-6851(85)90122-7;
RA   Simmons D.L., Hyde J.E., Mackay M., Goman M., Scaife J.;
RT   "Cloning studies on the gene coding for L-(+)-lactate dehydrogenase of
RT   Plasmodium falciparum.";
RL   Mol. Biochem. Parasitol. 15:231-243(1985).
CC   -!- CATALYTIC ACTIVITY: (S)-lactate + NAD(+) = pyruvate + NADH.
CC   -!- PATHWAY: Anaerobic glycolysis; final step.
CC   -!- SUBUNIT: Homotetramer.
CC   -!- SIMILARITY: Belongs to the LDH/MDH superfamily. LDH family.
CC   -----------------------------------------------------------------------
CC   Copyrighted by the UniProt Consortium, see http://www.uniprot.org/terms
CC   Distributed under the Creative Commons Attribution-NoDerivs License
CC   -----------------------------------------------------------------------
DR   EMBL; M14818; AAA29634.1; -; Genomic_DNA.
DR   GO; GO:0004459; F:L-lactate dehydrogenase activity; IEA:EC.
DR   InterPro; IPR011304; L_LDH_NAD.
DR   PROSITE; PS00064; L_LDH; PARTIAL.
PE   3: Inferred from homology;
KW   Glycolysis; NAD; Oxidoreductase.
FT   CHAIN        <1    >42       L-lactate dehydrogenase.
FT                                /FTId=PRO_0000168496.
FT   NON_TER       1      1
FT   NON_TER      42     42
SQ   SEQUENCE   42 AA;  4955 MW;  7FE28FDC341F474B CRC64;
     LLVYDNLLLN DKNEMNKDSS YDKKTNALDN YKNDSTIDME KK
//
```

UniParc This is the Uniprot archive. It contains a comprehensive and accurate history of protein sequences. This includes references to other databases for a given protein, as well as information about versions that have been removed from a database.

2.3 Other databases

There are many, many other useful databases. We will cover these during the class, concentrating on:

- Interpro and its constituents
- PDB, DSSP, FSSP, HSSP
- KEGG, EC and GO

2.4 SRS and Entrez

Entries in sequence databases[4] can be thought of consisting in two main parts: the sequence itself and the information about the sequence such as any unique identifiers assiged to it, what organism it is from, who deposited it in the database and where it is referred to by entries in other databases. The information about the data is called *metadata*.

Searching the metadata of entries in sequence and sequence-related databases can yield large amounts of information as well as sequence sets that can be used for further analysis. Two of the major interfaces allowing you to search the metadata of sequences (and carry out other analysis tasks as well) are the Sequence Retrieval Service (SRS) at the EBI and Entrez at the NCBI.

In the associated lecture and practical, we will focus on using SRS. We will look at Entrez in the context of blast searching of sequence databases later in the course.

[4] Wording in this section is for sequence databases, but the information is relevant for sequence-related databases also.

Chapter 3

Sequence alignment

3.1 Introduction

Aligning sequences allows us to judge the similarities and differences between two or more sequences. There are many pieces of software available for aligning sequences. There are some unifying themes underlying most of these programs, but there are also key differences that can greatly affect the results. Understanding the principles underlying sequence alignment, along with the impact of choices programmed into the software and choices you make when running the program, allows you to undertake meaningful analyses and interpret the results.

How do we choose between all the possible alignments between two sequences? A human would do this based on what they knew about the two sequences being aligned and using what they knew about evolution more generally. For example, if dealing with closely related peptide sequences, you might expect most amino acids in a sequence to line up with an identical amino acid in the other sequence. Where the residues are not identical, you might expect amino acids with similar characteristics to line up together more frequently than those that are very different.

How can we program computers to look for alignments that make sense from a biological perspective? We assign scores to each pair of elements that have been aligned. The score will be higher for pairs that are known to be found in "true" alignments frequently, and low for those that are not, and are thus considered less likely. So, for example, a tryptophan amino acid in one sequence paired with a tryptophan in another sequence, will get a high score; if it was paired with a tyrosine or phenylalanine, it would get a low score; if it was aligned with aspartate, it will get a very low score[1].

We use lists of scores[2] for all possible pairings that could arise in an alignment to calculate a total score for any given alignment of two sequences. We pick our

[1] If it is not immediately obvious to you why this is, take a look at your handouts for groupings of amino acids according to their properties.

[2] These lists are generally presented and used as matrices

best[3] alignment by choosing the one with the highest score.

3.2 Evolutionary models and alignment scoring

The reason to align sequences is usually to compare homologous sequences. As such, we would like to align the sequences to best represent evolutionary history. The scoring system we use should give higher scores to alignments that represent evolutionary history well than to alignments that do not.

In other words, models of the way sequences evolve form the basis of how distances between sequences are quantified and compared. These models are usually represented by matrices of scores of all the possible pairings that could occur in an alignment of two sequences. Choice of model, in practice this is the choice of the scoring matrix you use, will influence how sequences are aligned.

Most software gives you a choice of models to apply when aligning sequences[4].

Only a brief mention of some common models and scoring matrices are provided here. Further information is available from many introductory bioinformatics books. One you could try is [Higgs and Attwood, 2005].

3.2.1 Nucleic acid models

Jukes-Cantor model (JC). This model assumes that the four nucleotide bases occur equally frequently and that each has the same rate of substitution with every other base. Using these assumptions, a distance d is calculated from the number of sites that differ between the sequences D. This model involves a single parameter for the rate at which nucleotide substitutions occur.

Kimura two-parameter model (K2P). The frequencies of the four bases are assumed to be equal. However, this model involves two parameters: one for the rate of transitions and one for the rate of transversions. It has been observed that transitions are more common than transversions (i.e. A $< - >$ G, C $< - >$ T are more common.), and this model allows for this.

Hasegawa, Kishino and Yano model (HKY). This model allows for the four bases all having different frequencies as well as having separate parameters for the transition and transversion rates.

[3] Depending on the method used, this may or may not be *the* best possible alignment.

[4] The options available to you on the command line may far exceed those offered to you when using web-based forms. In addition, there are often more options on web-based forms than are visible by default. If you use web-based resources, it is worth investigating the default parameter settings, what these mean for you, and how much of the default behaviour you can modify. If you cannot use the web-based software to do what you want, investigate another way to run the program rather than persisting on doing less than ideal searches.

Note that all of the above models assume that time is reversible and that all the different sites in the sequence evolve at the same rate. The gamma distribution can be used to model the distribution of the rates found at different sites.

Choices of nucleotide substitution model is particularly important in phylogenetic analyses. However, this is beyond the scope of this course.

3.2.2 Amino acid models

In general terms, we need a 20 x 20 matrix containing scores to assign to each possible amino acid pairing in an alignment. There are two main model for amino acid evolution, the **PAM model** and the **BLOSUM model**. There are many variations on their themes.

Both PAM and BLOSUM scoring system are based on log-odds (lod) scores. Examples of two of the BLOSUM series of socring matrices are shown in figure 3.1 on page 15. Note that the scores provided in the matrices are scaled and then rounded to the nearest integer value[5]. This scaling does not affect what alignments are chosen, but make working with the scoring system much easier.

We will cover some of the details about amino acid scoring matrices during the class.

Specialist scoring matrices BLOSUM and PAM matrices are general purpose matrices. If the general characteristics of proteins of interest are quite different than the "average", then there are various options available to consider. For example, scoring matrices for the protein type of interest could be derived. Examples of this are the SLIM and PHAT matrices for transmembrane proteins [Ng et al., 2000, Muller et al., 2001].

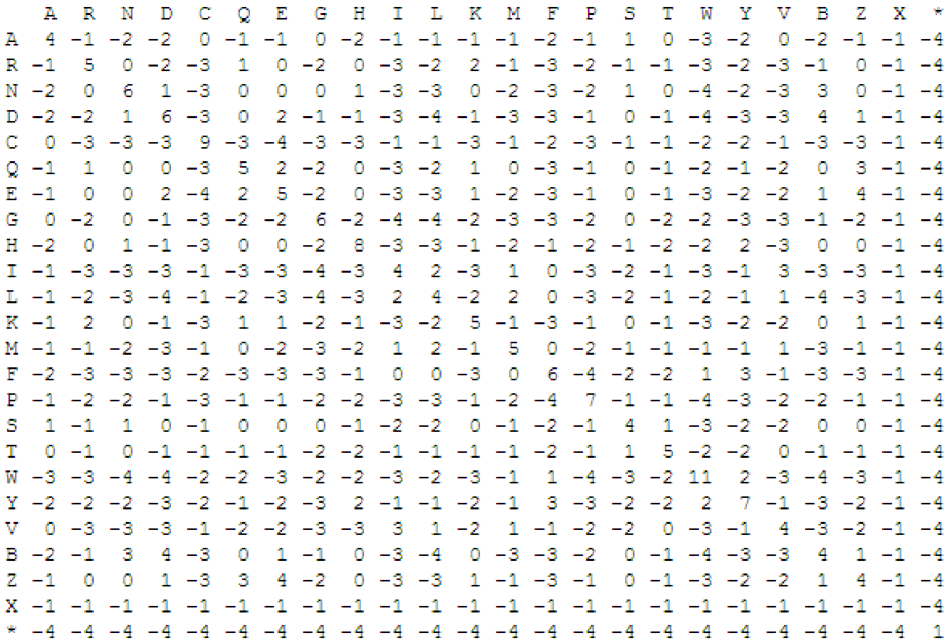
3.2.3 Gaps

Gaps are difficult to handle. We have to decide how to score them, and our choices greatly affect the results of alignment or sequence search programs, and do not necessarily reflect evolutionary biology. Some effects of gap choices will be illustrated in the section about the Clustalw alignment software.

In practical terms, gap opening and gap extension scores are not usually the same. If gaps are scored the same no matter where they are in a sequence, then we could get a situation where an alignment of two coding sequences that differ due to the insertion of a single gap of 9 nucleotides (or 3 amino acids) is scored the same as an alignment with 9 gaps inserted willy-nilly along the sequences. Which is more likely to have occurred from an evolutionary perspective? If we penalise heavily for opening a gap, but penalise relatively lightly for extending a gap that is already open, we would score the former alignment more favorably than the latter.

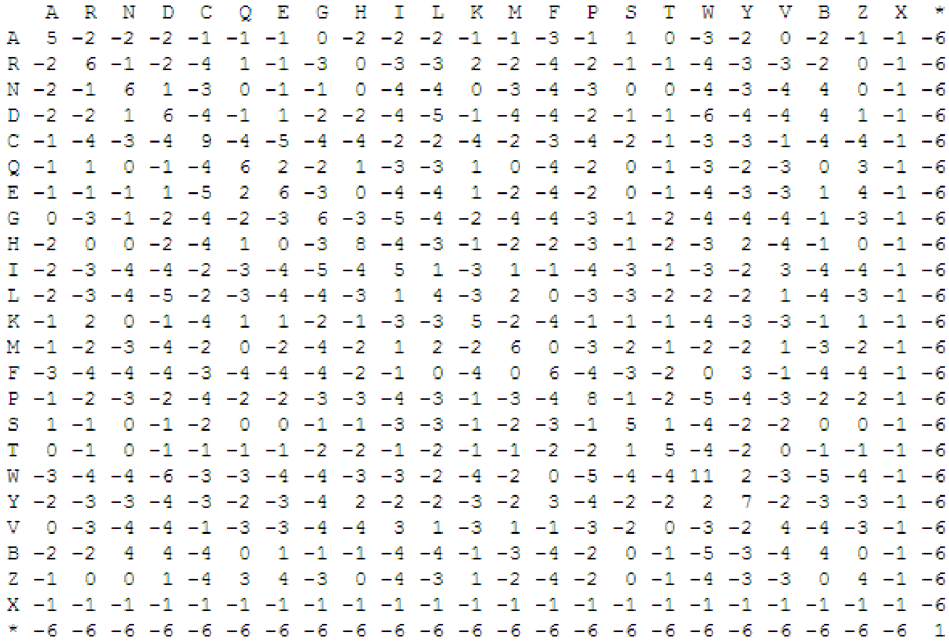
[5] I.e. the matrices do not contain the raw log-odds scores. This fact is useful for understanding some of the details involved in scoring sequence database search results.

Figure 3.1: A couple of examples of global and local alignment differences.



	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	B	Z	X	*
A	4	-1	-2	-2	0	-1	-1	0	-2	-1	-1	-1	-1	-2	-1	1	0	-3	-2	0	-2	-1	-1	-4
R	-1	5	0	-2	-3	1	0	-2	0	-3	-2	2	-1	-3	-2	-1	-1	-3	-2	-3	-1	0	-1	-4
N	-2	0	6	1	-3	0	0	0	1	-3	-3	0	-2	-3	-2	1	0	-4	-2	-3	3	0	-1	-4
D	-2	-2	1	6	-3	0	2	-1	-1	-3	-4	-1	-3	-3	-1	0	-1	-4	-3	-3	4	1	-1	-4
C	0	-3	-3	-3	9	-3	-4	-3	-3	-1	-1	-3	-1	-2	-3	-1	-1	-2	-2	-1	-3	-3	-1	-4
Q	-1	1	0	0	-3	5	2	-2	0	-3	-2	1	0	-3	-1	0	-1	-2	-1	-2	0	3	-1	-4
E	-1	0	0	2	-4	2	5	-2	0	-3	-3	1	-2	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
G	0	-2	0	-1	-3	-2	-2	6	-2	-4	-4	-2	-3	-3	-2	0	-2	-2	-3	-3	-1	-2	-1	-4
H	-2	0	1	-1	-3	0	0	-2	8	-3	-3	-1	-2	-1	-2	-1	-2	-2	2	-3	0	0	-1	-4
I	-1	-3	-3	-3	-1	-3	-3	-4	-3	4	2	-3	1	0	-3	-2	-1	-3	-1	3	-3	-3	-1	-4
L	-1	-2	-3	-4	-1	-2	-3	-4	-3	2	4	-2	2	0	-3	-2	-1	-2	-1	1	-4	-3	-1	-4
K	-1	2	0	-1	-3	1	1	-2	-1	-3	-2	5	-1	-3	-1	0	-1	-3	-2	-2	0	1	-1	-4
M	-1	-1	-2	-3	-1	0	-2	-3	-2	1	2	-1	5	0	-2	-1	-1	-1	-1	1	-3	-1	-1	-4
F	-2	-3	-3	-3	-2	-3	-3	-3	-1	0	0	-3	0	6	-4	-2	-2	1	3	-1	-3	-3	-1	-4
P	-1	-2	-2	-1	-3	-1	-1	-2	-2	-3	-3	-1	-2	-4	7	-1	-1	-4	-3	-2	-2	-1	-1	-4
S	1	-1	1	0	-1	0	0	0	-1	-2	-2	0	-1	-2	-1	4	1	-3	-2	-2	0	0	-1	-4
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-1	-1	-1	-2	-1	1	5	-2	-2	0	-1	-1	-1	-4
W	-3	-3	-4	-4	-2	-2	-3	-2	-2	-3	-2	-3	-1	1	-4	-3	-2	11	2	-3	-4	-3	-1	-4
Y	-2	-2	-2	-3	-2	-1	-2	-3	2	-1	-1	-2	-1	3	-3	-2	-2	2	7	-1	-3	-2	-1	-4
V	0	-3	-3	-3	-1	-2	-2	-3	-3	3	1	-2	1	-1	-2	-2	0	-3	-1	4	-3	-2	-1	-4
B	-2	-1	3	4	-3	0	1	-1	0	-3	-4	0	-3	-3	-2	0	-1	-4	-3	-3	4	1	-1	-4
Z	-1	0	0	1	-3	3	4	-2	0	-3	-3	1	-1	-3	-1	0	-1	-3	-2	-2	1	4	-1	-4
X	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-4
*	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	-4	1

Blosum62 Scoring Matrix



	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V	B	Z	X	*
A	5	-2	-2	-2	-1	-1	-1	0	-2	-2	-2	-1	-1	-3	-1	1	0	-3	-2	0	-2	-1	-1	-6
R	-2	6	-1	-2	-4	1	-1	-3	0	-3	-3	2	-2	-4	-2	-1	-1	-4	-3	-3	-2	0	-1	-6
N	-2	-1	6	1	-3	0	-1	-1	0	-4	-4	0	-3	-4	-3	0	0	-4	-3	-4	4	0	-1	-6
D	-2	-2	1	6	-4	-1	1	-2	-2	-4	-5	-1	-4	-4	-2	-1	-1	-6	-4	-4	4	1	-1	-6
C	-1	-4	-3	-4	9	-4	-5	-4	-4	-2	-2	-4	-2	-3	-4	-2	-1	-3	-3	-1	-4	-4	-1	-6
Q	-1	1	0	-1	-4	6	2	-2	1	-3	-3	1	0	-4	-2	0	-1	-3	-2	-3	0	3	-1	-6
E	-1	-1	-1	1	-5	2	6	-3	0	-4	-4	1	-2	-4	-2	0	-1	-4	-3	-3	1	4	-1	-6
G	0	-3	-1	-2	-4	-2	-3	6	-3	-5	-4	-2	-4	-4	-3	-1	-2	-4	-4	-4	-1	-3	-1	-6
H	-2	0	0	-2	-4	1	0	-3	8	-4	-3	-1	-2	-2	-3	-1	-2	-3	2	-4	-1	0	-1	-6
I	-2	-3	-4	-4	-2	-3	-4	-5	-4	5	1	-3	1	-1	-4	-3	-1	-3	-2	3	-4	-4	-1	-6
L	-2	-3	-4	-5	-2	-3	-4	-4	-3	1	4	-3	2	0	-3	-3	-2	-2	-2	1	-4	-3	-1	-6
K	-1	2	0	-1	-4	1	1	-2	-1	-3	-3	5	-2	-4	-1	-1	-1	-4	-3	-3	-1	1	-1	-6
M	-1	-2	-3	-4	-2	0	-2	-4	-2	1	2	-2	6	0	-3	-2	-1	-2	-2	1	-3	-2	-1	-6
F	-3	-4	-4	-4	-3	-4	-4	-4	-2	-1	0	-4	0	6	-4	-3	-2	0	3	-1	-4	-4	-1	-6
P	-1	-2	-3	-2	-4	-2	-2	-3	-3	-4	-3	-1	-3	-4	8	-1	-2	-5	-4	-3	-2	-2	-1	-6
S	1	-1	0	-1	-2	0	0	-1	-1	-3	-3	-1	-2	-3	-1	5	1	-4	-2	-2	0	0	-1	-6
T	0	-1	0	-1	-1	-1	-1	-2	-2	-1	-2	-1	-1	-2	-2	1	5	-4	-2	0	-1	-1	-1	-6
W	-3	-4	-4	-6	-3	-3	-4	-4	-3	-3	-2	-4	-2	0	-5	-4	-4	11	2	-3	-5	-4	-1	-6
Y	-2	-3	-3	-4	-3	-2	-3	-4	2	-2	-2	-3	-2	3	-4	-2	-2	2	7	-2	-3	-3	-1	-6
V	0	-3	-4	-4	-1	-3	-3	-4	-4	3	1	-3	1	-1	-3	-2	0	-3	-2	4	-4	-3	-1	-6
B	-2	-2	4	4	-4	0	1	-1	-1	-4	-4	-1	-3	-4	-2	0	-1	-5	-3	-4	4	0	-1	-6
Z	-1	0	0	1	-4	3	4	-3	0	-4	-3	1	-2	-4	-2	0	-1	-4	-3	-3	0	4	-1	-6
X	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-1	-6
*	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	-6	1

Blosum80 Scoring Matrix

3.3 Dynamic programming and heuristics

Algorithm, Dynamic programming and *heuristics* are terms you should be familiar with when considering sequence alignment software.

Algorithms are a set or ordered steps, essentially a recipe, for solving a problem. Different algorithms may be available to solve the same problem. If you were programming a computer to carry out the steps to solve your problem, you would want to consider who much memory and how much time the particular set of steps (i.e. the algorithm) you chose would take. For example, as your problem gets bigger, does the amount of computer time and/or memory grow linearly, polynomially, factorially, exponentially?

Dynamic programming is a way of solving problems where you make the best decision at each step along the way to your solution. With regards to alignment, this (in essence) allows you to figure out your best alignment by tracing a route through the options at each position in the alignment (i.e. match, mismatch, gap) that give you the highest score. There are many books that explain dynamic programming as it pertains to sequence alignment. For today, just remember that it is a way of solving problems that is good, but reasonably computationally intensive.

A heuristic is essentially a short cut that makes a problem tractable and that will give us a good solution most of the time. We cannot, however, prove that the solution we get is the best one. For example, in terms of sequence alignment, trying to search a large sequence database using a dynamic programming algorithm would take a lot of memory and a huge amount of time. Even a multiple sequence alignment for more than three or four sequences could take too long to carry out. So people design heuristics - reasonable shortcuts that (hopefully) provide reasonable answers. Of course, we need to remember that we took some short cuts when we interpret our results, especially when trying to figure out what happened if our results look strange.

Some heuristic algorithms, including the ones we will touch on when discussing multiple sequence alignments, are *greedy*. This means short-term decisions are made in solving the problem; hence the final solution will be heavily influenced by short-sighted decisions made early on[6].

Remember that in all cases, the optimal alignment between sequences will depend on the socring matrix and gap penalty system you use.

3.4 Pairwise alignment strategies

3.4.1 Global and local alignments

In brief, global alignment algorithms will align two sequences over both of their entire lengths. Local alignment algorithms will look for the best subalignments between two sequences. This may include the whole of each sequence,

[6] We will see this effect demonstrated using the Clustalw alignment program.

the whole of one of the sequences, or just parts of each sequence. Figure 3.2 on page 18 shows a couple of examples where a global alignment and a local alignment algorithm have been applied to pairs of sequences.

The two key algorithms for pairwise alignment are Needleman-Wunsch (global) and Smith-Waterman (local). Many programs implement some form of either of these[7].

Both global and local algorithms seek to find the highest scoring alignments. Some programs report only the highest scoring alignment, others will report multiple highest scoring alignments. If time permits, how scores are calculated for both algorithm types will be discussed in the class.

Needleman-Wunsch and Smith-Waterman algorithms employ dynamic programming. That is, they are guaranteed to find the highest scoring alignment. *Of course, the highest scoring alignment is not necessarily the correct one!*

Unless sequences are known to be homologous for their entire length, local alignment algorithms should be used.

3.4.2 Software for pairwise alignment

There are many programs that will do basic pairwise alignment. A few are listed below.

Global alignment (Needleman-Wunsch)

- needle
- stretcher
- lagan
- chaos

Local alignment (Smith-Waterman)

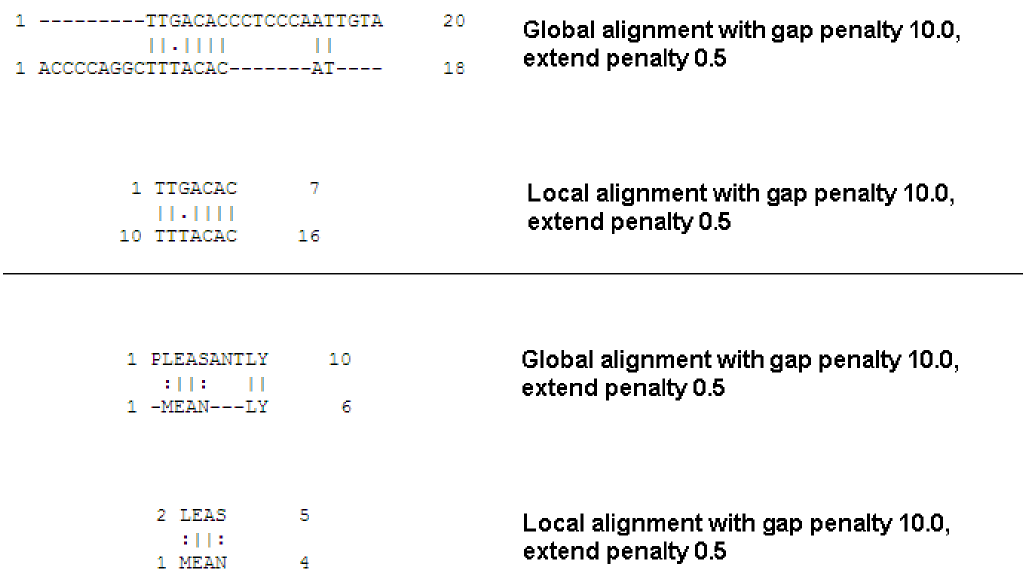
- water
- matcher
- ssearch
- spidey

Mixed global and local

- shuffle-lagan

[7] In your reading, if you see *Smith-Waterman*, think *local pairwise alignment*. Similarly, when you see *Needleman-Wunsch*, read *global pairwise alignment*

Figure 3.2: A couple of examples of global and local alignment differences.



```
 1 ---------TTGACACCCTCCCAATTGTA     20
            ||.||||       ||
 1 ACCCCAGGCTTTACAC-------AT----     18
```

Global alignment with gap penalty 10.0, extend penalty 0.5

```
 1 TTGACAC      7
   ||.||||
10 TTTACAC     16
```

Local alignment with gap penalty 10.0, extend penalty 0.5

```
1 PLEASANTLY     10
   :||:   ||
1 -MEAN---LY      6
```

Global alignment with gap penalty 10.0, extend penalty 0.5

```
2 LEAS      5
  :||:
1 MEAN      4
```

Local alignment with gap penalty 10.0, extend penalty 0.5

3.5 Multiple sequence alignment

Multiple sequence alignment is a key part of sequence-based bioinformatics. For example, a good alignment could illuminate conserved elements important to function, inform structural predictions, or allow phylogenetic tree estimation. As a corollary, if your alignment is bad, your downstream analyses will be meaningless...or worse, deceptive!

Even for a small number of sequences (e.g. 4 or more), a full dynamic programming solution to finding the optimal alignment would take far too long for most sequences. So, heuristic solutions have been devised. The most commonly used is *progressive alignment.*

We will consider progressive alignment algorithm as implemented by clustalw, as this is one of the most common multiple alignment programs. We will then comment briefly on some readily available software that may generate better alignments.

3.5.1 Clustalw - a case study of progressive alignment

Clustalw is a global alignment method. The general algorithm it employs is:

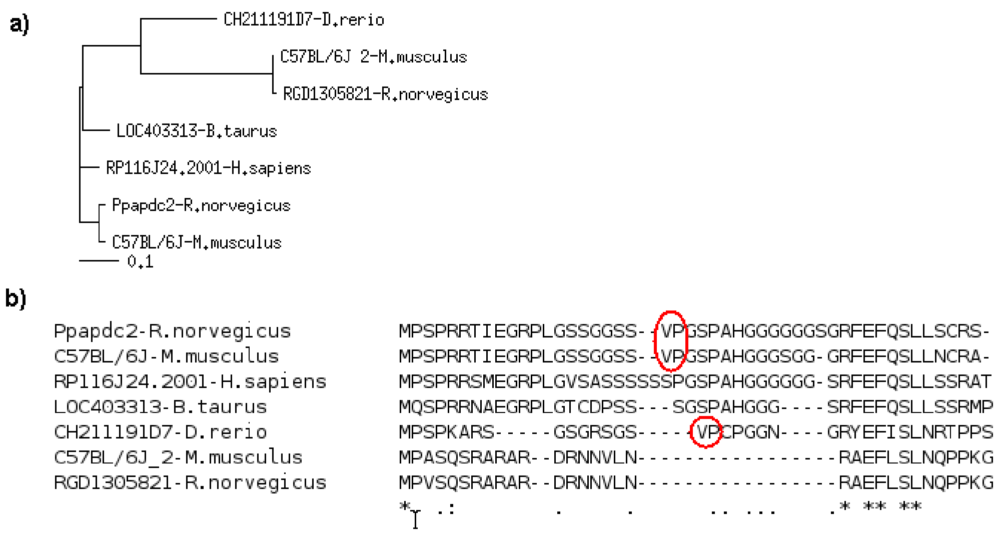
- Do a quick, global *pairwise* alignment of each pair of sequences in the set.
- Create a *quick, approximate* phylogenetic tree[8].
- Build the multiple sequence aligment by progressively aligning pairs of sequences according to their relationship in the approximate tree.

The above description misses out all the detail, but the general point is that the algorithm breaks down the multiple sequence alignment problem to a series of pairwise alignment problems. Note that because sequences are aligned in pairs *in the order specified by the approximate tree*, any alignment decisions made (e.g. the introduction of gaps) for closely related sequences, will be carried through to later stages of the alignment [9]. This is illustrated in figure 3.3 on page 20.

[8] **Important:** Do not use this as a phylogenetic tree!

[9] This is an example of a greedy algorithm.

Figure 3.3: Guide tree ordering effect on clustalw alignments. a) The guide tree produced for the sequences. b) The first section of the resulting alignment using Clustalw default settings. The red circles mark one place where the progressive alignment may perhaps have placed residues in a less than optimal position.

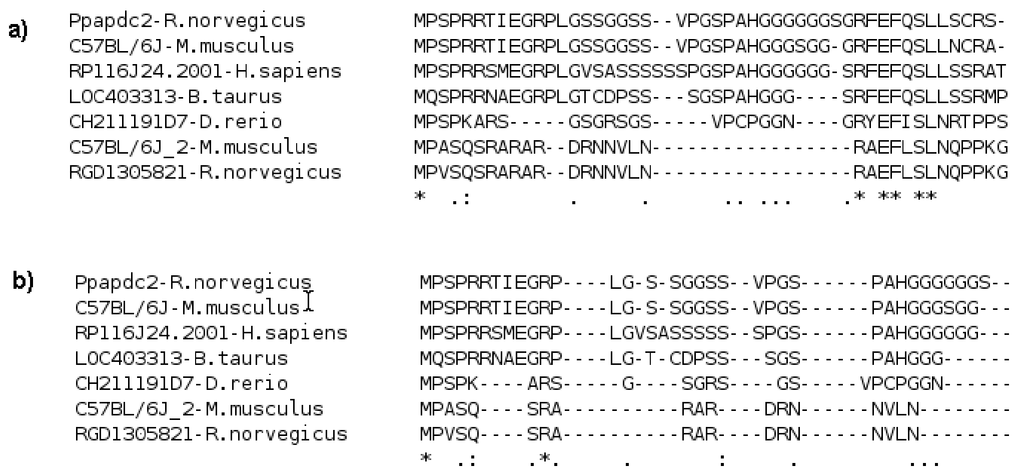


```
Ppapdc2-R.norvegicus       MPSPRRTIEGRPLGSSGGSS--VPGSPAHGGGGGGSGRFEFQSLLSCRS-
C57BL/6J-M.musculus        MPSPRRTIEGRPLGSSGGSS--VPGSPAHGGGSGG-GRFEFQSLLNCRA-
RP116J24.2001-H.sapiens    MPSPRRSMEGRPLGVSASSSSSSSPGSPAHGGGGGG-SRFEFQSLLSSRAT
LOC403313-B.taurus         MQSPRRNAEGRPLGTCDPSS---SGSPAHGGG----SRFEFQSLLSSRMP
CH211191D7-D.rerio         MPSPKARS-----GSGRSGS-----VPCPGGN----GRYEFISLNRTPPS
C57BL/6J_2-M.musculus      MPASQSRARAR--DRNNVLN-----------------RAEFLSLNQPPKG
RGD1305821-R.norvegicus    MPVSQSRARAR--DRNNVLN-----------------RAEFLSLNQPPKG
                           *  .:        .     .      .. ...    .* ** **
```

As mentioned earlier, the scoring matrix[10] and gap penalties chosen greatly affect the resulting alignments. This is illustrated in figure 3.4 on page 22.

[10] By default, Clustalw uses the Gonnet series of matrices. These were derived similarly to the PAM matrices, but based on more and more recent data. Note that The actual matrix used by Clustalw depends on how similar the sequences to be aligned at a given alignment step are.

Figure 3.4: Choice of gap pentalties can greatly affect the resulting alignment. Clustalw defaults for the pairwise and multiple alignment gap extension penalities, 0.1 and 0.2 respectively, were used for all alignments. a) Clustalw default gap opening penalty of 10 for pairwise and multiple alignments, b) gap opening penalty of 2 for pairwise and multiple alignments. All alignments used the Gonnet series of scoring matrix.



```
a)  Ppapdc2-R.norvegicus        MPSPRRTIEGRPLGSSGGSS--VPGSPAHGGGGGGSGRFEFQSLLSCRS-
    C57BL/6J-M.musculus         MPSPRRTIEGRPLGSSGGSS--VPGSPAHGGGSGG-GRFEFQSLLNCRA-
    RP116J24.2001-H.sapiens     MPSPRRSMEGRPLGVSASSSSSSPGSPAHGGGGGG-SRFEFQSLLSSRAT
    LOC403313-B.taurus          MQSPRRNAEGRPLGTCDPSS---SGSPAHGGG----SRFEFQSLLSSRMP
    CH211191D7-D.rerio          MPSPKARS-----GSGRSGS-----VPCPGGN----GRYEFISLNRTPPS
    C57BL/6J_2-M.musculus       MPASQSRARAR--DRNNVLN-----------------RAEFLSLNQPPKG
    RGD1305821-R.norvegicus     MPVSQSRARAR--DRNNVLN-----------------RAEFLSLNQPPKG
                                *  .:        .     .      .. ...    .* ** **

b)  Ppapdc2-R.norvegicus        MPSPRRTIEGRP----LG-S-SGGSS--VPGS------PAHGGGGGGS--
    C57BL/6J-M.musculus         MPSPRRTIEGRP----LG-S-SGGSS--VPGS------PAHGGGSGG---
    RP116J24.2001-H.sapiens     MPSPRRSMEGRP----LGVSASSSSS--SPGS------PAHGGGGGG---
    LOC403313-B.taurus          MQSPRRNAEGRP----LG-T-CDPSS---SGS------PAHGGG------
    CH211191D7-D.rerio          MPSPK----ARS-----G----SGRS----GS-----VPCPGGN------
    C57BL/6J_2-M.musculus       MPASQ----SRA----------RAR----DRN------NVLN--------
    RGD1305821-R.norvegicus     MPVSQ----SRA----------RAR----DRN------NVLN--------
                                *  .:    .*.     .       :     .         ...
```

Key issues to look at when deciding whether an automatically generated multiple alignment is reasonable include whether functional motifs line up and whether gaps have been introduced in regions likely or unlikely to contain them. If you have access to secondary structure information, this can also be used to manually improve the alignment.

The global nature of clustalw means that you should only use this program on sequences of similar length. You may need to edit your sequences so the are of similar length before proceeding, or you may need to consider programs using other algorithms more suitable for what you are trying to do.

3.5.2 Other multiple sequence alignment software

There are many other multiple sequence alignment algorithms implemented. The most common programs use variations of progressive alignment, but they differ in attributes like whether and on what basis an initial guide tree is produced, how many rounds of progressive alignment are undertaken, whether there are alignment refinement stages, what scoring systems are used, and so on. The choices made for any given piece of software affects how well it aligns sequences, and particularly, how well it may align sequences with particular characteristics. The choices also affect how long the program takes to run.

Some software to look into when you come to generating your own sequences include:

- Muscle for general peptide multiple sequence alignment [Edgar, 2004] .
- T-coffee and M-coffee for general peptide multiple sequence alignment[Notredame et al., 2000, Wallace et al., 2006].
- Lagan and MLagan for aligning long homologous sequences [Brudno et al., 2003].

3.5.3 Profile alignment

We will discuss sequence profiles and their alignment briefly during the course. In terms of accuracy, profile-profile alignments tend to be more accurate than profile-sequence alignments, which are in turn, more accurate than sequence-sequence alignments.

A key software package to consider when working with profiles is Hmmer [Eddy, 1998].

3.5.4 Alignment Editors

As will hopefully be obvious by now, you will often have to work on an aligment manually before using the alignment as the basis for downstream analyses. Some programs including sequence alignment editors are listed below.

- Mega [Tamura et al., 2007]

- Jalview [Clamp et al., 2004]
- Cinema [Lord et al., 2002]
- Seaview [Galtier et al., 1996]

Chapter 4

Searching sequence-based databases

4.1 Introduction

We looked at searching sequence annotation earlier. Here we will look at a variety of search programs that are used to search for similar sequences in sequence databases. If time permits, profile searching software and profile databases will be discussed during the class.

The programs we look at today are available via the web. However, many of them can be installed on your local machine to search databases held on your local machine or held remotely. We will discuss the pros and cons of using remote and local services and databases during the course.

There are many types of sequence-based searches you could consider doing. For example:

- Search a sequence databases for similar sequences
- Search a profile or pattern database with a sequence
- Search a sequence database with a pattern or profile

Underlying all database searching programs is the idea of pattern matching. The differences between programs comes down to the types of patterns you are searching with, the types of patterns you are searching against, and the way the programs go about finding matches between them. Most database searching software will include some statistical measure of the significance of matches found.

4.2 Health warnings

Database searching should be approached like an experiment. You should define your aims before your start. This will save you an enormous amount of time, both in terms of working on the computer with your sequence, and also when trying to bring together and report your findings later.

Before you start searching with a sequence, it is useful to outline answers to questions like the following:

- What am I trying to find out/What do I want to do with the results?
- What kind of database do I want to search with my sequence? E.g. nucleotide, protein, pattern, profile?
- Which database(s) in particular do I want to search? Why?
- Are there are any subsets of the database that I might want to restrict my search to?
- Do I want to take into account potential frameshifts in my coding sequences?
- What format is my sequence in[1]?
- Do I want to filter my sequence for repeats and low complexity regions before searching?
- Is the scoring system I've chosen appropriate?

4.3 Some terminology and considerations

4.3.1 Terms

Sensitivity The ability to detect true matches. The most sensitive search may retrieve many true positives, but could also return many false positives.

Specificity The ability to find only true matches. The price for a very specific search could be that some true matches will be missed.

Query sequence The sequence you are searching with.

Subject sequence The sequence you are comparing your query sequence to. You could think of a sequence database as made up of many subject sequences.

Word In the context of sequence comparisons, a number of consecutive residues. E.g. a word size of 11 for nucleotides means a group of 11 consecutive nucleotides in the sequence.

Ktup A synonym for *word* in the context of sequence comparison. Used particularly with the programs in the Fasta package.

[1] The issue of sequence formats is no longer the problem it once was. Many programs are capable of dealing with different sequence formats. For programs that cannot, there are a number of format translators. A good one is **seqret** available as part of the EMBOSS package.

E-value A statistical score, based on the extreme value distribution, used to interpret the results of database searches. In general terms, E-values indicate the number of alignments expected by chance when carrying out a search just like the one you just did. This is discussed in further detail later in this chapter. For small E values (e.g. < 0.001), this is the same as the probability of finding an alignment as good as the one you found in a search like the one you just did by random chance.

HSP A High scoring Segment Pairs are segments of your sequence that match a sequence in the database. Such segments are given a score, and if that score exceeds a set value, the program attempts to find longer matching areas in this pair of sequences.

MSP A Maximal-scoring Segment Pair is the segment pair with the highest score in a pairwise comparison.

Raw score The score of an alignment, calculated as the sum of substitution and gap scores. Because of differences between scoring matrices, raw scores are not always directly comparable.

Bit score Bit scores are derived from the raw alignment score, but are normalized with respect to the scoring system. Bit scores can be used to compare alignment scores from different searches. [2]

Scoring matrix A matrix containing the scores that are given to pairs of amino acids or nucleotides when aligning them.

4.3.2 Searching sequences with sequences - generalities

There is software available that will allow you to do a Smith-Waterman local pairwise alignment of your sequence of interest with every other sequence in a database. However, this can be very slow. A variety of software has been written to make searching sequence databases with other sequences much faster. This speed is attained through the use of "heuristic" algorithms. The choice of algorithm affects the speed of the program, and also the quality of the results. In many cases you can set the values of parameters used during the search, which will also affect the results returned.

The most commonly used heuristic algorithms for searching sequence databases are Fasta[Pearson, 1990], and Blast (including the closely related psi-blast)[Altschul et al., 1997].

4.3.3 Filtering and Masking

Nucleotide and amino acid sequences often contain low-complexity regions. These are relatively uninformative regions such as short tracts of repeats like proline-rich regions or poly-A tails, Alu sequences in humans, etc. Searching

[2] Specifically, they are raw scores that have been converted from the log base of the scoring matrix that creates the alignment to log base 2.

using sequences containing such regions is likely to return many non-related, yet high scoring sequences.

Filtering/masking can improve searches.

Programs have been devised to *filter* unwanted segments of sequence from within a larger sequence. Generally, these programs identify low complexity regions in the query sequence and *replace them* with a series of N's (in the case of nucleotide sequences), or X's (in the case of peptide sequences). One N or one X replaces each residue in the region.

Masking. In contrast to filtering, word masks do not alter the sequence itself. Rather, they cause certain portions of the query sequence to be skipped during the neighborhood word generation step of the BLAST algorithm[3]. Here, the query sequence is kept intact when generating comprehensive alignments, but these alignments are seeded by neighborhood word hits involving only the more informative, unmasked regions of the sequence.

If you filter a query sequence, your score will be reduced[4] It is recommended that you start out using soft masking, where the filtering is applied during the word seeingn stage, but not used during the extension phase[5].

4.4 Fasta

The Fasta package contains a number of database searching programs. On the surface, these are similar to the more widely-used blast family of programs, but there are key differences in the implementations of the two packages. Current versions of Blast should yield similar sensitivity and specificity to searches using FASTA, and will run much faster. However, you may find hits using FASTA that are not found in a Blast search and vice versa.

In addition, there are programs available within the Fasta package that have functionality that is not replicated exactly among the Blast family of programs. Examples of this are the fastx and fasty programs, making allowances for gaps and frameshifts when comparing nucleotides to peptide databases.

Due to time constraints, we will not be able to consider the Fasta package further during this course.

[3] The way Blast works will be described during the class.

[4] For example, in default versions of blast for amino acid searches, amino acids paired with a masked position are given a fixed negative score. Filtered query sequences can also result in the loss of some alignments because HSPs may be broken into smaller HSPs, which might drop the score too low for that match to be kept.

[5] These phases are explained in section 4.5

4.5 Blast

4.5.1 How Blast works

A general outline of how blast works will be covered during the class.

4.5.2 Blast releases

There are two main releases of blast: NCBI Blast and Wu-Blast. For many, the NCBI blast release is more familiar. Wu-blast is faster [Korf et al., 2003] and offers other advantages[6]. The majority of publicly available blast servers offer NCBI rather than Wu-blast, but both are available at the EBI.

4.5.3 Blast flavours

Different *flavours* of blast are used for different types of searches. See table 4.1.

Table 4.1: Blast flavours

Flavour	Query type	Subject type(db type)
blastn	nucleotide	nucleotide
blastp	peptide	peptide
blastx	nucleotide	peptide
tblastn	peptide	nucleotide
tblastx	nucleotide	nucleotide

Blastx, tblastn and tblastx all take a nucleotide query or subject and translate it on the fly for the purposes of the search.

In addition to what might be called the *standard* flavours above, there are also newer incarnations of blast for various types of tasks. See table 4.2[7].

An overview of which blast programs to consider for different tasks is given on the NCBI Blast program selection guide page.

4.5.4 Filtering in blast

.

Whether to filter or not depends on your sequence and your aims in searching the database. You will need to check what the defaults are for any given search program. Below is a list of what you can expect for blast at the moment in general terms.

[6] For example, it has more command-line parameters for advanced users, allowing them to fine-tune their searches.

[7] There are many other blast derivatives not listed here.

Table 4.2: Blast derivatives

Name	Purpose	Query type	Subject type
megablast	Like blastn, but optimised to find nearly identical matches quickly. Works best when seqs >95% similar.	nucleotide	nucleotide
discontiguous megablast	Like megablast but aimed at cross-species queries [8]. Uses seeds where mismatches are allowed.	nucleotide	nucleotide
psiblast	Iterative program to search for weak peptide similarities.	peptide (profile)	peptide
phiblast	Pattern-hit initiated blast - searches only those members of a peptide database containing a user-specified motif.	peptide query and peptide motif	peptide
rpsblast	Reverse psiblast searches a query sequence against a database of profiles	peptide	peptide profiles

- **Filtering is turned on or off by default differently in different NCBI blast programs.** For example, currently, web-based blastp has filtering off by default, while web-based tblastn has filtering turned on.
- **Filtering is turned off** when using wu-blast on the command line.
- NCBI blastall uses SEG for filtering with peptide sequences (e.g. blastp, blastx, tblastn, tblastx).
- NCBI blastall uses DUST for filtering with nucleotide sequences (e.g. blastn, megablast)

4.5.5 Results formatting for web-based NCBI blast

There are many formatting options available both via the command line and web-based blast. Here we focus on those available via the NCBI's web interface for blast[9].

CDS feature Annotated sequence features in or around the hit are displayed within the result. For query sequences not in a public database, the sequence is translated using the CDS translation on the matching database sequence as a guide. Mismatches are highlighted in pink.

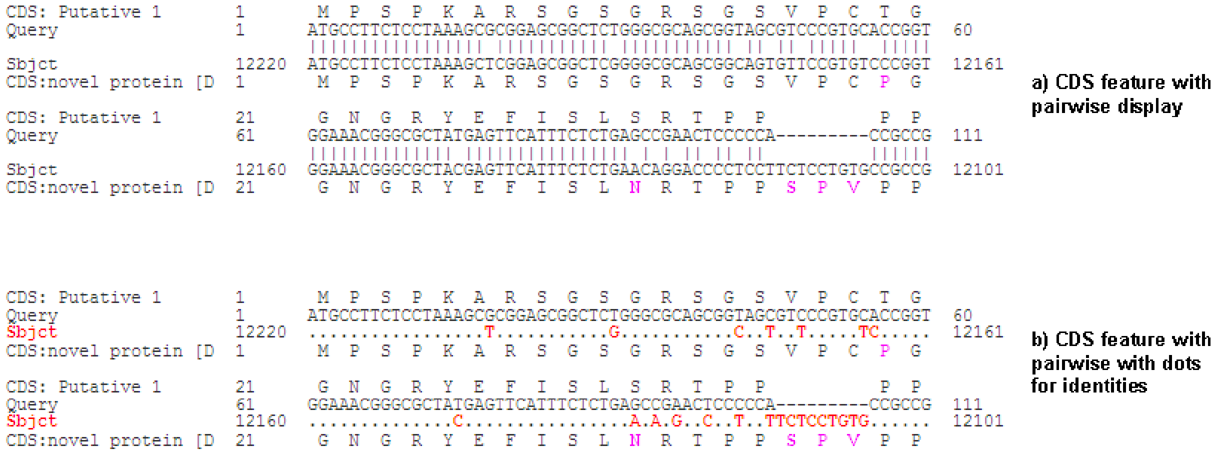
Pairwise view The default view.

[9] The web resources at the NCBI are under constant development. The following notes are true as of October, 2007.

Pairwise with dots for identities view Like pairwise, but easier to see where there are differences in the alignments.

See figure 4.1 on page 31 for examples of CDS feature turned on with pairwise and with pairwise with dots for identities views.

Figure 4.1: CDS feature with a)pairwise and b)pairwise with dots for identities views.



```
CDS: Putative 1       1        M  P  S  P  K  A  R  S  G  S  G  R  S  G  S  V  P  C  T  G
Query                 1       ATGCCTTCTCCTAAAGCGCGGAGCGGCTCTGGGCGCAGCGGTAGCGTCCCGTGCACCGGT  60
                              ||||||||||||||||| ||||||||||| ||||||||||| || || |||||  |||||
Sbjct                 12220   ATGCCTTCTCCTAAAGCTCGGAGCGGCTCGGGGCGCAGCGGCAGTGTTCCGTGTCCCGGT  12161
CDS:novel protein [D  1        M  P  S  P  K  A  R  S  G  S  G  R  S  G  S  V  P  C  P  G

CDS: Putative 1       21       G  N  G  R  Y  E  F  I  S  L  S  R  T  P  P           P  P
Query                 61      GGAAACGGGCGCTATGAGTTCATTTCTCTGAGCCGAACTCCCCCA---------CCGCCG  111
                              |||||||||||||| |||||||||||||||| | | || || ||          ||||||
Sbjct                 12160   GGAAACGGGCGCTACGAGTTCATTTCTCTGAACAGGACCCCTCCTTCTCCTGTGCCGCCG  12101
CDS:novel protein [D  21       G  N  G  R  Y  E  F  I  S  L  N  R  T  P  P  S  P  V  P  P
```

a) CDS feature with pairwise display

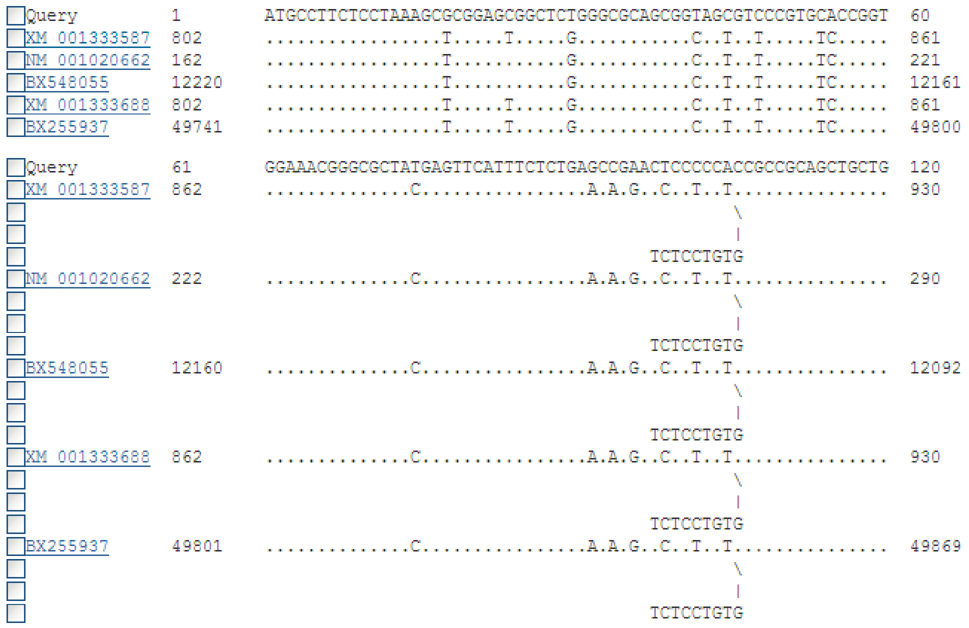
```
CDS: Putative 1       1        M  P  S  P  K  A  R  S  G  S  G  R  S  G  S  V  P  C  T  G
Query                 1       ATGCCTTCTCCTAAAGCGCGGAGCGGCTCTGGGCGCAGCGGTAGCGTCCCGTGCACCGGT  60
Sbjct                 12220   .................T...........G...........C..T..T.....TC.....  12161
CDS:novel protein [D  1        M  P  S  P  K  A  R  S  G  S  G  R  S  G  S  V  P  C  P  G

CDS: Putative 1       21       G  N  G  R  Y  E  F  I  S  L  S  R  T  P  P           P  P
Query                 61      GGAAACGGGCGCTATGAGTTCATTTCTCTGAGCCGAACTCCCCCA---------CCGCCG  111
Sbjct                 12160   ..............C................A.A.G..C..T..TTCTCCTGTG......  12101
CDS:novel protein [D  21       G  N  G  R  Y  E  F  I  S  L  N  R  T  P  P  S  P  V  P  P
```

b) CDS feature with pairwise with dots for identities

Query-anchored view A stacked view of the hits aligned to the query sequence. This is a good view to choose if looking for things like SNPs or amino acid substitutions among related sequences.

Query-anchored view with dots for identities Similar to the query-anchored view, but easier to spot differences between the sequences. See figure 4.2 on page 33 for examples of query-anchored view with dots for identities.

Figure 4.2: Query-anchored with dots for identities view.



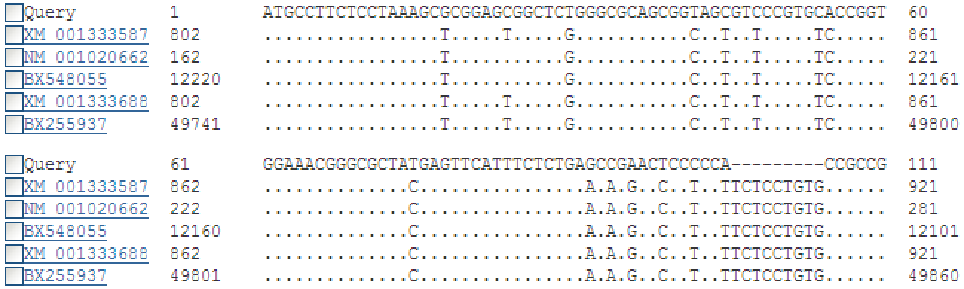
```
□Query          1      ATGCCTTCTCCTAAAGCGCGGAGCGGCTCTGGGCGCAGCGGTAGCGTCCCGTGCACCGGT  60
□XM 001333587   802    .................T.....T.....G...........C..T..T.....TC.....  861
□NM 001020662   162    .................T...........G...........C..T..T.....TC.....  221
□BX548055       12220  .................T...........G...........C..T..T.....TC.....  12161
□XM 001333688   802    .................T.....T.....G...........C..T..T.....TC.....  861
□BX255937       49741  .................T.....T.....G...........C..T..T.....TC.....  49800

□Query          61     GGAAACGGGCGCTATGAGTTCATTTCTCTGAGCCGAACTCCCCCACCGCCGCAGCTGCTG  120
□XM 001333587   862    ..............C................A.A.G..C..T..T...............  930
□                                                                   \
□                                                                   |
□                                                           TCTCCTGTG
□NM 001020662   222    ..............C................A.A.G..C..T..T...............  290
□                                                                   \
□                                                                   |
□                                                           TCTCCTGTG
□BX548055       12160  ..............C................A.A.G..C..T..T...............  12092
□                                                                   \
□                                                                   |
□                                                           TCTCCTGTG
□XM 001333688   862    ..............C................A.A.G..C..T..T...............  930
□                                                                   \
□                                                                   |
□                                                           TCTCCTGTG
□BX255937       49801  ..............C................A.A.G..C..T..T...............  49869
□                                                                   \
□                                                                   |
□                                                           TCTCCTGTG
```

Flat query-anchored view A stacked view of the hits aligned to the query sequence with insertions and deletions presented as they are in most text-based multiple alignment program output.

Flat query-anchored view with dots for identities Similar to the flat query-anchored view, but easier to spot differences between the sequences.

See figure 4.3 on page 35 for examples of query-anchored view with dots for identities.

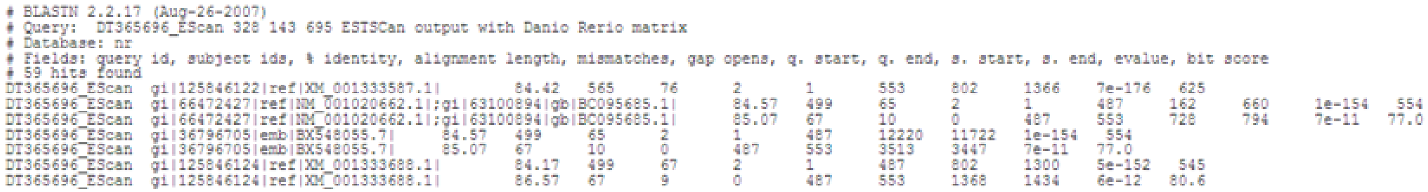
Figure 4.3: Query-anchored with dots for identities view.



```
□Query         1       ATGCCTTCTCCTAAAGCGCGGAGCGGCTCTGGGCGCAGCGGTAGCGTCCCGTGCACCGGT  60
□XM 001333587  802     .................T.....T.....G...........C..T..T.....TC.....  861
□NM 001020662  162     .................T...........G...........C..T..T.....TC.....  221
□BX548055      12220   .................T...........G...........C..T..T.....TC.....  12161
□XM 001333688  802     .................T.....T.....G...........C..T..T.....TC.....  861
□BX255937      49741   .................T.....T.....G...........C..T..T.....TC.....  49800

□Query         61      GGAAACGGGCGCTATGAGTTCATTTCTCTGAGCCGAACTCCCCCA---------CCGCCG  111
□XM 001333587  862     ..............C................A.A.G..C..T..TTCTCCTGTG......  921
□NM 001020662  222     ..............C................A.A.G..C..T..TTCTCCTGTG......  281
□BX548055      12160   ..............C................A.A.G..C..T..TTCTCCTGTG......  12101
□XM 001333688  862     ..............C................A.A.G..C..T..TTCTCCTGTG......  921
□BX255937      49801   ..............C................A.A.G..C..T..TTCTCCTGTG......  49860
```

Hit table Summary of results in a table. Useful for exporting into other programs. See figure 4.4 on page 37 for an example.

Figure 4.4: Example of hit table results.



```
# BLASTN 2.2.17 (Aug-26-2007)
# Query:  DT365696_EScan 328 143 695 ESTSCan output with Danio Rerio matrix
# Database: nr
# Fields: query id, subject ids, % identity, alignment length, mismatches, gap opens, q. start, q. end, s. start, s. end, evalue, bit score
# 59 hits found
DT365696_EScan  gi|125846122|ref|XM_001333587.1|        84.42   565     76      2       1       553     802     1366    7e-176   625
DT365696_EScan  gi|66472427|ref|NM_001020662.1|;gi|63100894|gb|BC095685.1|      84.57   499     65      2       1       487     162     660     1e-154   554
DT365696_EScan  gi|66472427|ref|NM_001020662.1|;gi|63100894|gb|BC095685.1|      85.07   67      10      0       487     553     728     794     7e-11   77.0
DT365696_EScan  gi|36796705|emb|BX548055.7|     84.57   499     65      2       1       487     12220   11722   1e-154   554
DT365696_EScan  gi|36796705|emb|BX548055.7|     85.07   67      10      0       487     553     3513    3447    7e-11   77.0
DT365696_EScan  gi|125846124|ref|XM_001333688.1|        84.17   499     67      2       1       487     802     1300    5e-152   545
DT365696_EScan  gi|125846124|ref|XM_001333688.1|        86.57   67      9       0       487     553     1368    1434    6e-12   80.6
```

4.5.6 Scoring in NCBI blast

When you search a sequence database, you may find many hits. We want a scoring system that helps us to differentiate the sequence hits that are related to the original sequence, from those that share similarities by chance. The scoring system Blast uses is designed to give high scores to sequences that are hopefully homologues of the original sequence, and lower scores to sequences that are not. The scoring system is based on a variety of assumptions and uses heuristics. Understanding how blast works and the implications of the parameters you set when you run a search, will enable you to devise good searching plans and intelligently interpret your results.

Fear ye not the math. The equations below will be explained during the class.

1. The Karlin-Altschul equation:

   $$E = kmne^{-\lambda S}$$

   where **E** is the *number* of alignments scoring at least S expected by chance during a database search like the one you just did[10], **m** is the number of residues in the query sequence, **n** is the number of residues in the database[11]. **k** and λ are constants[12]**S** is the raw score.[13].

2. The *normalised* score in nats[14] of *an individual* HSP.:

   $$S'_{nats} = \lambda S - ln(k)$$

   This equation calculates a normalised score for *a single* HSP.

3. The equation below calculates the sum score for HSPs when more than one HSP is found for the query and a particular sequence in the database[15]. The equation sums the bit scores for each HSP, normalises the score and carries out some corrections explained after the equation.

   $$S'_{sum} = \lambda \sum_{i=1}^{r} S_r - rln(kmn) + ln(r!)$$

   Here, **r** is the number of HSPs for a given hit.

[10] This means a database search of a sequence the length of yours against a database the size you searched, using the scoring matrix you used.

[11] The m and n values actually used in blast scoring are corrected for edge effects.

[12] λ is a scaling factor used for converting raw scores to bit scores. k is usually around 0.1 and corrects for the fact that the optimal local alignment scores for two different starting points in a pair of sequences can be highly correlated. For example, if there is a high scoring alignment starting at residues 3,3, then it is likely that there will also be a high scoring alignment starting at 4,4.

[13] From this equation, it can be seen that E values will increase as the database gets bigger and decrease as the score gets bigger (i.e. as the alignment gets longer or stronger). Recall: small E-values are hopefully indicative of homologous sequences.

[14] Bits are values converted to base 2 logarithms. Nats are values converted to base *e*. In blast, λ is reported in nats, but bit scores, as their name implies, are reported in bits.

[15] This is just one of the sum score equations utilised by NCBI blast.

λ contributes to normalising the sum score, as does the ln(k) section of the second term. The first term in the equation involves the sum of all the individual HSP raw scores (in bits). Just summing all the raw scores overestimates the score that should be assigned to a set of HSPs, so the second term includes a correction factor for the overestimation. The final term is essentially a bonus added to the score if multiple HSPs are found ordered between query and subject sequences.

4. Using the sum statistic, an aggregated pairwise P-value for that sum score can be calculated[16].

$$P_r = \frac{e^{-S_{sum}} S^{r-1}}{r!(r-1)!}$$

5. Pair-wise, test-corrected (for multiple HSPs) sum-P value

$$P'_r = \frac{P_r}{\beta_{r-1}(1-\beta)}$$

where β is a gap decay constant.

6. Corrected E value for sum statistics

$$Expect(r) = \frac{effective\_database\_length}{n} * P'_r$$

Here, effective database length is essentially the length of your query sequence times the length of the database, with both these values having had corrections applied to them. The n in the denominator is the length of the subject sequence in the alignment.

7. Equations linking P and E values

$$P = 1 - e^{-E}$$

$$E = -ln(1-P)$$

Note that for very small values (e.g. < 0.001), E and P are practially identical.

Scores imply a particular evolutionary distance

Different scoring matrices are designed to help determine homologous sequences at different evolutionary distances. In other words, by choosing a scoring system, you are implicitly choosing a particular evolutionary distance (in other words, a particular level of difference between your sequences).

To keep things simple, we can illustrate this by considering a particular nucleotide scoring scheme and a bit of simple algebra.

[16] The equation given is for NCBI Blast. This may be done slightly differently in Wu-blast.

Remember the log-odds equation presented during the class:

$$s(a, b) = 1/\lambda * ln(p_{ab}/f_a f_b)$$

Let's say you want to find sequences which are around 88% similar to your own. We can easily calculate scores that will optimise the search to find matches with this level of similarity.

We make a few (big) assumptions to make our calculations easy:

- Assume that all mismatch types occur with equal probability
- Assume that the background composition of both the query and subject sequences involve equal proportions of each nucleotide (i.e. 25% each).

Using the above, we know that frequencies of each nucleotide is 0.25.

We are optimizing for 88% matches, which, under our assumptions, we have a target frequency[17] of 0.22 of each type of identical match. With 88% matches, we have 12% mismatches. There are 12 types of mismatch possible, so we have a target frequency of 0.01 for each type of mismatch.

That is, we have $p_{ab} = 0.22$ for a match, $p_{ab} = 0.01$ for a mismatch, and $f_a = f_b = 0.25$. Plugging this into the log-odds equation, we get the following for a match:

$$s(a, b) = 1/\lambda * log(0.22/(.25)^2)$$

and the following for a mismatch:

$$s(a, b) = 1/\lambda * log(0.01/(.25)^2)$$

This gives $\frac{1.26}{\lambda}$ for a match and $\frac{-1.83}{\lambda}$ for a mismatch.

Remember that λ is just an arbitrary scaling factor. So if we scale using $\lambda = 0.25$, and then round off, we get a scoring system of +5/-7.

Why not just guess a scoring system and keep trying until you find one that works?

- The scoring system you choose implies a particular ideal target frequency. Isn't it better to know what this is and plan your blast experiments accordingly?
- Do you have so much excess time on your hands to just play around randomly with blast parameters?
- Multiple testing....are there implications here[Frommlet et al., 2004]?

[17] To understand what we mean by target frequency here, think about tossing a coin. Tossing a fair coin, we believe that on average we will see heads 50% of the time and tails 50% of the time. That is, if we toss the coins many times, the proportion of heads will be 0.5 and the proportion of tails will be 0.5. Here, our target frequency for heads is 0.5 and for tails is 0.5. Because of the assumptions we've laid out, we can say our target frequency for a given matching nucleotide pair will be 0.22.

What are the implied target frequencies for some of the default parameters for Blast?

You need to use the log-odds equation again, this time solving for λ. This is not as straightforward as the math done above.

To generate the results in table 4.3, I used the program available as part of the supplementary data for [Eddy, 2004][18].

Table 4.3: Target percent identities for some nucleotide scoring systems

Program	Default match	Default mismatch	Target % identity
NCBI blastn, command line	+1	-3	99%
Wu-blastn, command line	+5	-4	65%
NCBI blastn, NCBI website	2	-3	89%
Wu-blastn, EBI website	2	-3	89%

Note that the word sizes and gap costs can also differ between Blast incarnations, even on the same website. These factors can also affect whether particular hits are found or not.

4.5.7 Recent Blast scoring developments

Conditional compositional score matrix adjustment

This is a relatively new development for scoring amino acid blast searches. It is turned on by default using web-based blastp at the NCBI, and turned off by default for web-based tblastn at the NCBI[19].

When set, the following occurs:

- Blast is executed with using the chosen scoring matrix.
- For sequences that pass one of three criteria (length ratio < 3.0, compositional distance < 0.16, compositional angle < 70 degrees), and where alignments with a prelimary E-value $< 100 * maximum(E)$, progress to second stage, which consists of:
  - Scoring matrix adjusted
  - Query and subject re-aligned
  - New E-value calculated[20].

[18] You can also use this software to generate target frequencies for amino acids.

[19] Using this choice rarely changes *which* sequences appear in the output but it can change their statistical scores

[20] E-values are not recalculated if not improved by new scoring.

The scoring option **Universal compositional matrix adjustment** carries out the same steps as the conditional compositional matrix adjustment, but does so for any hit where the with a prelimary E-value $< 100 * maximum(E)$.

Useful references for blast and blast scoring include:

Books

- Korf et al. [2003]

Papers

- Altschul et al. [1997]
- Altschul et al. [2005]
- Eddy [2004]
- Schaffer et al. [2001]
- Yu et al. [2006]

4.6 Searching with profiles

Due to time constraints, we will not be able to cover the use of profiles in database searching, even though it is a very important topic. A list of a lot of alignment software, including sequence search software and profile searching software is available on the wikipedia site: `http://en.wikipedia.org/wiki/Sequence_alignment_software`.

Chapter 5

Basic tasks in bioinformatics

5.1 Examples of basic tasks

A few examples of basic common tasks when dealing with sequence data are given below. There are many other common tasks and the general message is the same: *If it seems like something other people have probably had to do too, then there is probably a tool already available to do it.*

5.1.1 Sequence formatting

The format of a sequence refers to the way it appears as you see it on the page. Many programs can now handle a variety of sequence formats, but some still cannot. Common format names include fasta, embl, genbank, clustal, phylip...therea are many others. An example of the same sequence in fasta, embl and genbank format is shown in figure 5.1 on page 44.

There are programs designed to convert sequences between different formats, so there is no need to attempt this by hand or to attempt to write a program.

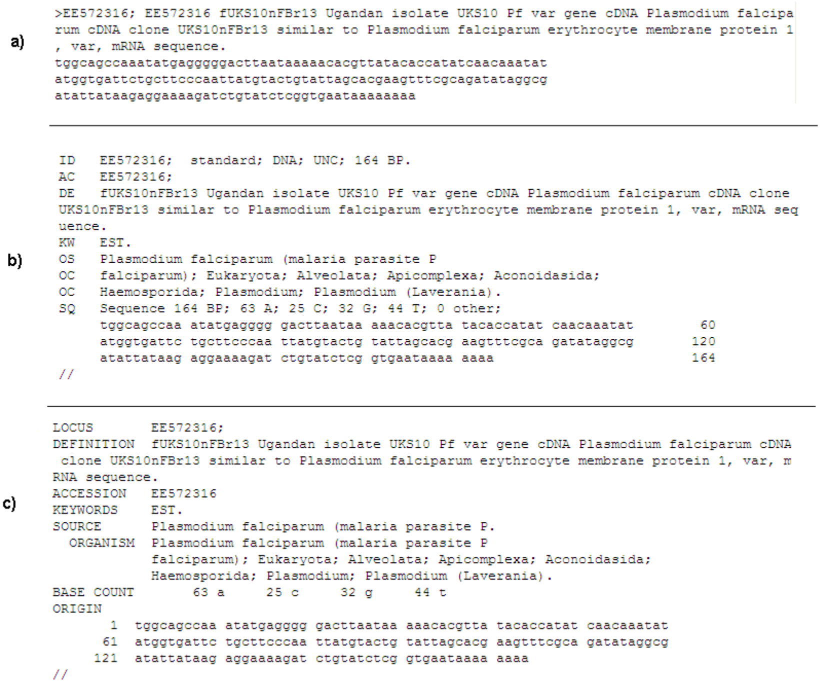
For example, most commonly used sequence formats are recognised and can be converted using the EMBOSS program **seqret** or the program **Readseq**. Both these programs are available to download for local use [1].

5.1.2 Finding ORFs and/or genes

There are many tools out there for finding ORFs or finding genes. These tasks are different, with the latter a much more complex issue than the former. A list of some available gene finding software is available on the genefinding.org site. A couple of simple programs for generating translations of coding sequence will be run during the practical session.

[1] Readseq is a stand-alone program whereas. Seqret is available as part of the EMBOSS package.

Figure 5.1: A sequence in a) fasta format, b) Embl format (feature table not included) and c) Genbank format (feature table not included).



```
>EE572316; EE572316 fUKS10nFBr13 Ugandan isolate UKS10 Pf var gene cDNA Plasmodium falcipa
rum cDNA clone UKS10nFBr13 similar to Plasmodium falciparum erythrocyte membrane protein 1
, var, mRNA sequence.
tggcagccaaatatgaggggacttaataaaaacacgttatacaccatatcaacaaatat
atggtgattctgcttcccaattatgtactgtattagcacgaagtttcgcagatataggcg
atattataagaggaaaagatctgtatctcggtgaataaaaaaaa
```

```
ID   EE572316;  standard; DNA; UNC; 164 BP.
AC   EE572316;
DE   fUKS10nFBr13 Ugandan isolate UKS10 Pf var gene cDNA Plasmodium falciparum cDNA clone
UKS10nFBr13 similar to Plasmodium falciparum erythrocyte membrane protein 1, var, mRNA seq
uence.
KW   EST.
OS   Plasmodium falciparum (malaria parasite P
OC   falciparum); Eukaryota; Alveolata; Apicomplexa; Aconoidasida;
OC   Haemosporida; Plasmodium; Plasmodium (Laverania).
SQ   Sequence 164 BP; 63 A; 25 C; 32 G; 44 T; 0 other;
     tggcagccaa atatgagggg gacttaataa aaacacgtta tacaccatat caacaaatat        60
     atggtgattc tgcttcccaa ttatgtactg tattagcacg aagtttcgca gatataggcg       120
     atattataag aggaaaagat ctgtatctcg gtgaataaaa aaaa                        164
//
```

```
LOCUS       EE572316;
DEFINITION  fUKS10nFBr13 Ugandan isolate UKS10 Pf var gene cDNA Plasmodium falciparum cDNA
 clone UKS10nFBr13 similar to Plasmodium falciparum erythrocyte membrane protein 1, var, m
RNA sequence.
ACCESSION   EE572316
KEYWORDS    EST.
SOURCE      Plasmodium falciparum (malaria parasite P.
  ORGANISM  Plasmodium falciparum (malaria parasite P
            falciparum); Eukaryota; Alveolata; Apicomplexa; Aconoidasida;
            Haemosporida; Plasmodium; Plasmodium (Laverania).
BASE COUNT       63 a     25 c     32 g     44 t
ORIGIN
        1  tggcagccaa atatgagggg gacttaataa aaacacgtta tacaccatat caacaaatat
       61  atggtgattc tgcttcccaa ttatgtactg tattagcacg aagtttcgca gatataggcg
      121  atattataag aggaaaagat ctgtatctcg gtgaataaaa aaaa
//
```

5.1.3 Vector marking and clipping

If you are manually looking for vector sequence and marking it or deleting it in your own sequence files, then stop! There are a number of programs out there designed to do just this. Examples include cross_match and vectorstrip. Vector clipping is as easily done on many sequences as a single sequence using the right software (or using software in the right way.) If time permits, the Staden software will be demonstrated, where sequence will be quality checked, vector will be marked and the cleaned sequences assembled. Using a system like this, hundreds of sequences can be processed in only a few moments.

5.1.4 PCR primer design

Again, this is a common task and there is a variety of software available. One popular piece of software, available for the command line or via the web is **primer3**. An EMBOSS wrapper to this called **eprimer3** is also available.

5.1.5 Graphical viewing of data and annotation

This area will be mentioned during the lectures. There is much software available and it is worth investigating this area before starting your work to find tools that will enable you to work more effienciently and present your data more effectively.

Chapter 6

General tips

6.1 Computing related

- If you have just thought of it, there is probably software to help you achieve it.
- There is probably software to help you achieve things you hadn't even thought of.
- Software is constantly being developed. Have a look around for new tools.
- Read the documentation!
- Understand the documentation!
- Garbage in, garbage out.
- Software is not an answer, it is a tool. It is easy to use the wrong tool for a job. Most software will provide you with an answer. It doesn't mean the answer is meaningful.
- If you find yourself doing a task over and over, there will be ways to automate it.

Don't use a piece of software just because and only because that's what the guy next to you in the lab uses. Maybe that guy really knows what he's doing. But maybe he doesn't. What do you think? It is worth finding out a bit more about what that software does, how it does it and what else is out there.

6.2 Sequence formatting

- As mentioned earlier, there are programs capable of converting sequences and alignments from one format to another. A good program to try first for reformatting tasks is the EMBOSS program **seqret**.

- Small letters and capitals letters are not always interchangeable. Some programs may infer that you are less sure of sequence tracts recorded in small letters than you are of tracts in capital letters. It is safest to store sequences with capital letters if you aren't sure whether the software you use will infer anything by the case[1].
- If software has good error checking, it will report an error if you try to run the program on a sequence with formatting it doesn't understand. *Not all software is so helpful.* Some software will generate meaningless results, which at their worst are deceptive, if you give them data in the wrong format. Beware.

6.3 Filenames

There are easy rules of thumb for naming your files:

- **Give your file meaningful names.** Ideally you should know what is in the file without having to open it.
- **Use only standard characters for your filenames.** Don't use any funny characters in your filenames such as

  ```
  ? ! | ( ) { } % #
  ```

  and so on. If you do, you are just asking for trouble.
- Use standard filename extensions. E.g. *.fasta* for fasta files *.aln* for clustalw multiple sequence alignment files and so on. Some software won't recognize sequences with a different file extension[2], and you will save a lot of time if you know what format your sequence is in without having to open the file to check.
- **Don't use spaces in your filenames.** If your data files are sent to you from somewhere else with spaces already in the names, rename them without spaces. One idea is to replace spaces with underscores _ There are rare instances where underscores are a problem...but they are rare.
- Use an automated method to rename files if you are renaming more than a few. There are simple command line tools available for Linux (e.g. try the *rename* command), and a number of graphical tools available on Windows (e.g The Bulk Rename Utility.)
- **Name relevant files by date order.** For some types of files, it may be worth naming them so they appear in date order when you browse the list. For example, perhaps you run a particular blast search every week. If you save the file to something like YearMonthDay_mySeq_DB.blastX (e.g. 20071031_mySeq_nr.blastn), then your files will sort according to the date you ran the searches.

[1] There are many ways to convert sequences from small to capital letters. One choice is to use the command line option -supper in the EMBOSS program seqret

[2] Crappy but true.

- Sequence names may need to be short, may need to be altered to be unique within the first N letters, or may need to be changed if spaces are included. E.g. by default clustalw understands sequence names of less than 30 characters[3]. A space in the sequence name denotes the end of the sequence name, so the following will not be recognised as unique:
  - R. norvegicus Ppapdc2
  - R. norvegicus RGD1305821

  Here, both sequences are understood to have the name **R.**. The command line version of clustalw returns a meaningful enough message for you to figure out the problem. Web servers usually just report that there aren't any sequences in your file even though there are.

6.4 Data standards and data submission

6.4.1 Standards and ontologies

Entries in databases have generally been annotated by humans using the words and phrases they deemed best to describe their data. Sometimes this includes good descriptions, sometimes there is almost no description. The terms they chose might be the same ones you would choose, or they might be quite different. And did they mean the same thing you mean when they used a particular word or phrase to describe their data?

The diversity of annotation in terms of quantity and quality has led to a large push for data standards and ontologies. In essence, standards describe *what* types of information should be included about a type of data, while ontologies dictate *how* information should be conveyed, that is, what words or terms should be used to describe a particular thing or meaning.

The most well known standard is arguably the Minimum Information about a Microarray Experiment [Brazma et al., 2001]. Another initiative, the Genomics Standards Consortium, is working on developing a standard for the richer annotation of genomics and metagenomic data sets. There are many other standards or standard recommendations. However, some them can be hard to track down. This issue prompted the establishment of the MIBBI project. MIBBI is a one-stop shop for so-called *minimum information* lists as well as providing a space for developers to discover and discuss their and other initiatives.

There are also initiatives to create ontologies that researchers can use to annotate their data such that others will understand the same thing that was meant. The Open Biomedical Ontologies site is a good place to start looking for these.

[3] This is more than the default used to be set to, and different sites can choose different sequence name lengths, so it best to err on the side of cation and use shorter rather than longer names.

6.4.2 Data submission

If you produce digital biological data and want to publish it, chances are you will need to submit it to a public repository. There are currently large, international repositories for most sequence-based data and also microarray and proteomic data. Others are being developed for additional types of data such as metabolomic.

There are usually guidelines published by the respository about what is required in order to submit your data. For established data types such as sequence data, specialised software can help you submit. It is best to look into these issues early. Much hair-pulling frustration can be caused by attempting to annotate a large experiment and submit it at the last minute!

6.5 Local versus remote, web-based software

Running jobs via web pages (or web services) has its pros and cons. These are just a few.

Pros

- Programs accessed via a web page are often easy to run.
- Programs not available locally may be offered via web pages or services.
- If a powerful server is running the software, then the speed may be good.
- Many databases may be accessible through web-based software.
- In many cases, results may be pretty and may have hyperlinks to other relevant information.

Cons

- You rely on someone else to provide what you need.
- You are releasing your data onto someone else's machine.
- The resource could be down when you need it, or removed completely.
- There is often a restriction on the number of analyses you can run. (e.g. 400 blasts per day at NCBI)
- There is often no way to automate large numbers of searches.
- You may have many more configuration options locally.
- All the useful results files may not be returned to you (e.g. the guide tree for clustalw alignments is often not provided.)
- Running software on remote machines can be slow.

6.6 Finding programs and documentation

How do you find the programs you need? An incomplete list of ideas is below.

- Look for software on national and international bioinformatics sites such the NCBI, the EBI and the bioinformatics.ca links directory.
- Try the Bio-Linux bioinformatics documentation system.
- Try searching in Pubmed for key terms to try and find articles on new software.
- Browse the contents lists for bioinformatics journals.
- Google searches sometimes unearth programs of interest.

6.7 Keeping track, keeping results and keeping up

6.7.1 Keeping track

You should keep notes about your *in silico* experiments, just like you do with lab-based experiments. Details to keep track of include:

- What is your analysis plan? (Why are you doing what you are doing?)
- What programs are you using? What version are you running (e.g. version number and website if applicable)?
- What databases are you accessing? Where are they stored, what version are they?
- Where have you stored your results and what are they called?

You have seen that parameter settings can greatly affect alignment scores and the types of results you get when searching databases. You don't want to tell your supervisor (or worse, the whole world) about some great finding if you then cannot replicate the finding later. All it might take would be to forget your parameter settings, or for an updated version of a database to have slightly different information in it. If you had the details recorded, it would be easy to track the problem.

6.7.2 Keeping results

Many websites have implemented tools to help you log your activities and results, at least in the short term. For example:

My NCBI is an account you can set up at the NCBI to save searches, set up e-mail alerts for new content, to display links to web resources and to set up filters to apply to your search results. Once you are signed into My NCBI, you can also choose to have your search terms highlighted in PubMed output. You can also save results in Collections. You are allowed up to 1500 results in a given collection and 100 collections.

SRS and Ensembl offer similar facilities to My NCBI. In the case of Ensembl accounts, you can also set up group permissions so you can work collaboratively.

6.7.3 Keeping up

New and email alerts There are news and alert services for all sorts of things. For example, most journals have RSS feeds, as do the NCBI, EBI and many other sites. There are many email alert facilities such as Google Alerts, BioMed central email alerts , or even alerts when new sequences similar to ones you are interested in are added to a database, such as swiss-shop.

Connotea and cite-u-like .Connotea and cite-u-like are two on-line reference management services aimed at scientists. Using them you can store your own references, tag them, and also discover publications others have stored and tagged.

Bibliography

S. F. Altschul, T. L. Madden, A. A. Schaffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman. Gapped blast and psi-blast: a new generation of protein database search programs. *Nucleic Acids Res*, 25(17):3389–402, 1997. Lm05110/lm/nlm Journal Article Research Support, U.S. Gov't, P.H.S. Review England.

S. F. Altschul, J. C. Wootton, E. M. Gertz, R. Agarwala, A. Morgulis, A. A. Schaffer, and Y. K. Yu. Protein database searches using compositionally adjusted substitution matrices. *Febs J*, 272(20):5101–9, 2005. Z01 lm000072-10/lm/nlm Journal Article Review England.

A. Brazma, P. Hingamp, J. Quackenbush, G. Sherlock, P. Spellman, C. Stoeckert, J. Aach, W. Ansorge, C. A. Ball, H. C. Causton, T. Gaasterland, P. Glenisson, F. C. Holstege, I. F. Kim, V. Markowitz, J. C. Matese, H. Parkinson, A. Robinson, U. Sarkans, S. Schulze-Kremer, J. Stewart, R. Taylor, J. Vilo, and M. Vingron. Minimum information about a microarray experiment (miame)-toward standards for microarray data. *Nat Genet*, 29(4): 365–71, 2001. Journal Article Research Support, Non-U.S. Gov't Research Support, U.S. Gov't, P.H.S. United States.

M. Brudno, C. B. Do, G. M. Cooper, M. F. Kim, E. Davydov, E. D. Green, A. Sidow, and S. Batzoglou. Lagan and multi-lagan: efficient tools for large-scale multiple alignment of genomic dna. *Genome Res*, 13(4):721–31, 2003. NISC Comparative Sequencing Program Comparative Study Evaluation Studies Journal Article Research Support, Non-U.S. Gov't Research Support, U.S. Gov't, Non-P.H.S. Research Support, U.S. Gov't, P.H.S. United States.

M. Clamp, J. Cuff, S. M. Searle, and G. J. Barton. The jalview java alignment editor. *Bioinformatics*, 20(3):426–7, 2004. Journal Article England.

S. R. Eddy. Where did the blosum62 alignment score matrix come from? *Nat Biotechnol*, 22(8):1035–6, 2004. Evaluation Studies Journal Article Review United States.

S. R. Eddy. Profile hidden markov models. *Bioinformatics*, 14(9):755–63, 1998. R01 hg01363/hg/nhgri Journal Article Research Support, Non-U.S. Gov't Research Support, U.S. Gov't, P.H.S. Review England.

R. C. Edgar. Muscle: multiple sequence alignment with high accuracy and high throughput. *Nucleic Acids Res*, 32(5):1792–7, 2004. Comparative Study Evaluation Studies Journal Article England.

F. Frommlet, A. Futschik, and M. Bogdan. On the significance of sequence alignments when using multiple scoring matrices. *Bioinformatics*, 20(6):881–7, 2004. Comparative Study Evaluation Studies Journal Article Research Support, Non-U.S. Gov't Validation Studies England.

N. Galtier, M. Gouy, and C. Gautier. Seaview and phylo_win: two graphic tools for sequence alignment and molecular phylogeny. *Comput Appl Biosci*, 12 (6):543–8, 1996. Journal Article Research Support, Non-U.S. Gov't England Cabios.

Paul G. Higgs and Teresa K. Attwood. *Bioinformatics and molecular evolution.* Blackwell, Malden, Mass. ; Oxford, 2005. 2004021066 Paul G. Higgs and Teresa K. Attwood. ill. ; 25 cm. Includes bibliographical references and index.

Ian Korf, Mark Yandell, Joseph Bedell, and Stephen Altschul. *BLAST.* O'Reilly, Sebastopol, Calif. ; Farnham, 2003. GB A3-Y7706 Ian Korf, Mark Yandell, and Joseph Bedell ; [foreword by Stephen Altschul]. ill. ; 24 cm. "An essential guide to the Basic Local Alignment Search Tool"-Cover. Includes bibliographical references and index.

P. W. Lord, J. N. Selley, and T. K. Attwood. Cinema-mx: a modular multiple alignment editor. *Bioinformatics*, 18(10):1402–3, 2002. Journal Article Research Support, Non-U.S. Gov't England.

T. Muller, S. Rahmann, and M. Rehmsmeier. Non-symmetric score matrices and the detection of homologous transmembrane proteins. *Bioinformatics*, 17 Suppl 1:S182–9, 2001. Journal Article England.

P. C. Ng, J. G. Henikoff, and S. Henikoff. Phat: a transmembrane-specific substitution matrix. predicted hydrophobic and transmembrane. *Bioinformatics*, 16(9):760–6, 2000. Journal Article Research Support, U.S. Gov't, Non-P.H.S. Research Support, U.S. Gov't, P.H.S. England.

C. Notredame, D. G. Higgins, and J. Heringa. T-coffee: A novel method for fast and accurate multiple sequence alignment. *J Mol Biol*, 302(1):205–17, 2000. Journal Article Research Support, Non-U.S. Gov't England.

W. R. Pearson. Rapid and sensitive sequence comparison with fastp and fasta. *Methods Enzymol*, 183:63–98, 1990. Comparative Study Journal Article United states.

A. A. Schaffer, L. Aravind, T. L. Madden, S. Shavirin, J. L. Spouge, Y. I. Wolf, E. V. Koonin, and S. F. Altschul. Improving the accuracy of psi-blast protein database searches with composition-based statistics and other refinements. *Nucleic Acids Res*, 29(14):2994–3005, 2001. Journal Article Review England.

K. Tamura, J. Dudley, M. Nei, and S. Kumar. Mega4: Molecular evolutionary genetics analysis (mega) software version 4.0. *Mol Biol Evol*, 24(8):1596–9, 2007. Journal Article Research Support, N.I.H., Extramural Research Support, Non-U.S. Gov't United States.

I. M. Wallace, O. O'Sullivan, D. G. Higgins, and C. Notredame. M-coffee: combining multiple sequence alignment methods with t-coffee. *Nucleic Acids Res*, 34(6):1692–9, 2006. Evaluation Studies Journal Article Research Support, Non-U.S. Gov't England.

Y. K. Yu, E. M. Gertz, R. Agarwala, A. A. Schaffer, and S. F. Altschul. Retrieval accuracy, statistical significance and compositional similarity in protein sequence database searches. *Nucleic Acids Res*, 34(20):5966–73, 2006. Evaluation Studies Journal Article Research Support, N.I.H., Intramural England.