Core Practicals in Bioinformatics

This series of Core Practicals is organised into three self-directed practicals, written as a step-by-step manual. While all tasks can be completed on any computer with access to the internet and the free software package MEGA, two three-hour contact sessions are dedicated to the Core Practicals.

In each practical, you will find the corresponding **learning objectives**, and several **self-assessment questions** that are designed to help you understand the materials better. Answer keys are provided at the end of each practical.

This 33-page document consists of the following three core pracs:

Practical 1: Introduction to NCBI and Ensembl (pages 2–12)

Practical 2: Sequence alignment and similarity searching (pages 13–23)

Practical 3: Sequence analysis and phylogenetics (pages 24–33)

Data, **Supplementary Notes** and **Videos** that are relevant to each of these practicals are available on Blackboard (in ***BINF6000 Practicals > Core Practicals***). You are strongly encouraged to go through this manual and the additional information before or when attending the contact sessions in Week 1 and Week 2.

IMPORTANT

- You will be assessed based on the materials covered in the Core Practicals in **Assessment 1** (due 14:00 Brisbane time, Friday 17 March 2023)
- This assessment is an online multiple-choice quiz (15%) administered on Blackboard.

Practical 1: Introduction to NCBI and Ensembl

This practical consists of two parts:
PART 1. Getting to know NCBI (5 Sections)
PART 2. Browsing genome maps using Ensembl (3 Sections)

Learning objectives:

PART 1
1A. Access NCBI resources
1B. Search for sequences using NCBI
1C. Do advanced NCBI searches
1D. Download sequences from GenBank
1E. Integrate information from PubMed

PART 2
2A. Access the Ensembl Genome Browser
2B. Examine genomic regions in ContigView
2C. View synteny between the human and mouse genomes

Practical Considerations:

When saving sequence files it is important that you use a **plain text editor**, e.g. *Notepad* on PC, and *TextEdit* on Mac. Word-processing programs like Microsoft Word or Google Docs adopt Rich Text Format (RTF) which is not optimal for bioinformatics research, as it introduces hidden line-break characters and/or unwanted formatting margins into the file; these might not be interpreted correctly by most bioinformatics programs.

Additional references:

- Textbook – Zvelebil NJ & Baum JO (2008) *Understanding Bioinformatics,* Garland Science: *Chapter 3*: Dealing with databases, p45-68.
- See also **Practical 1 Supplementary Notes** and **Videos** on Blackboard

PART 1. Getting to know NCBI

The National Center for Biotechnology Information (NCBI) is an important online resource for bioinformatics and genomics databases and tools. NCBI houses many publicly available biological databases, with data types such as DNA sequences (GenBank), gene expression (GEO), and literature (PubMed). NCBI also has a range of tools for querying and analysing this data. Researchers publishing studies using genetic or protein data are generally required to submit these data to public repositories, many of which are hosted by NCBI. Thus, NCBI is an invaluable resource for researchers to share, find and analyse the huge amount of data that is generated every year within the biological sciences.

The tasks in PART 1 draw on some of the most commonly used NCBI resources to illustrate some basic procedures in bioinformatics. **§ Supplementary Notes 1–3**

1A. Access NCBI resources

1. Navigate to http://www.ncbi.nlm.nih.gov - this is the main NCBI entry portal.

2. Spend some time and familiarise yourself with the website. Have a look at all available database resources on NCBI.

The easiest way to navigate around NCBI is the cross-database search using the search bar on top of the page. You can search All Databases at once by default or restrict a search to specific database (from the pull-down menu).

Q1. GenBank is part of the International Sequence Database Collaboration comprising which three organisations?

Q2. Reference Sequence (RefSeq) is a collection of curated, highly redundant genomic DNA, transcript (RNA), and protein sequences produced by NCBI. True or False?

Q3. Which database at NCBI contains raw sequencing reads of RNA and DNA from high-throughput technologies?

3. Do a simple search of your last name across all databases on NCBI.

On the search results page, you will see the different databases grouped into categories (literature, genes, proteins…) with the number of hits in each. Unless you have an uncommon name, you should see hits across different databases – usually because researchers with your last name have submitted data to NCBI or its linked databases.

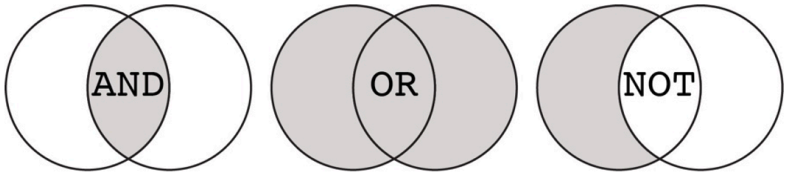
1B. Searching for sequences in NCBI's databases

Case Study
Who? *Salmonella enterica* serovar Typhimurium (aka *Salmonella* serovar Typhimurium).
Why? They cause food poisoning (gastroenteritis).
What? SopE, a type III secretion protein (*effector*); they have an *effect* (usually deleterious) on the host cell.
So what? Some bacterial effectors are known to exert their effects on eukaryote cells. SopE protein mimics the Rho-family guanine exchange factor (GEF) in human.

1. Query **`effector`** in NCBI's search bar. How many hits do you get in the **Nucleotide** database?

Boolean operators (**AND**, **NOT** and **OR**) can be used refine your search.
A AND B = the intersection of **A** and **B**.
A OR B = both **A** and **B** sets, inclusive of the data shared by both.
A NOT B = everything in **A** that is not in **B**.

FIGURE 1



2. Query **`SopE effector`** in the Nucleotide database. Now try three different searches with either **AND**, **OR**, or **NOT** between the search terms and see how it affects your results.

Q4. Which Boolean function (**AND**, **OR** or **NOT**) is used by default between the terms when no function is specified?

3. Enclose the words in quotes e.g. **`"SopE gene"`** to search using an exact phrase.

Q5. How many sequence records in the Nucleotide database that contain:
(a) the phrase "**SopE gene**"? (b) the words **SopE** and **gene**?

4. Query **`SopE`** in the Nucleotide database. You should see something similar to **Figure 2**. Note that NCBI updates their web interface frequently, so the exact layout may differ.

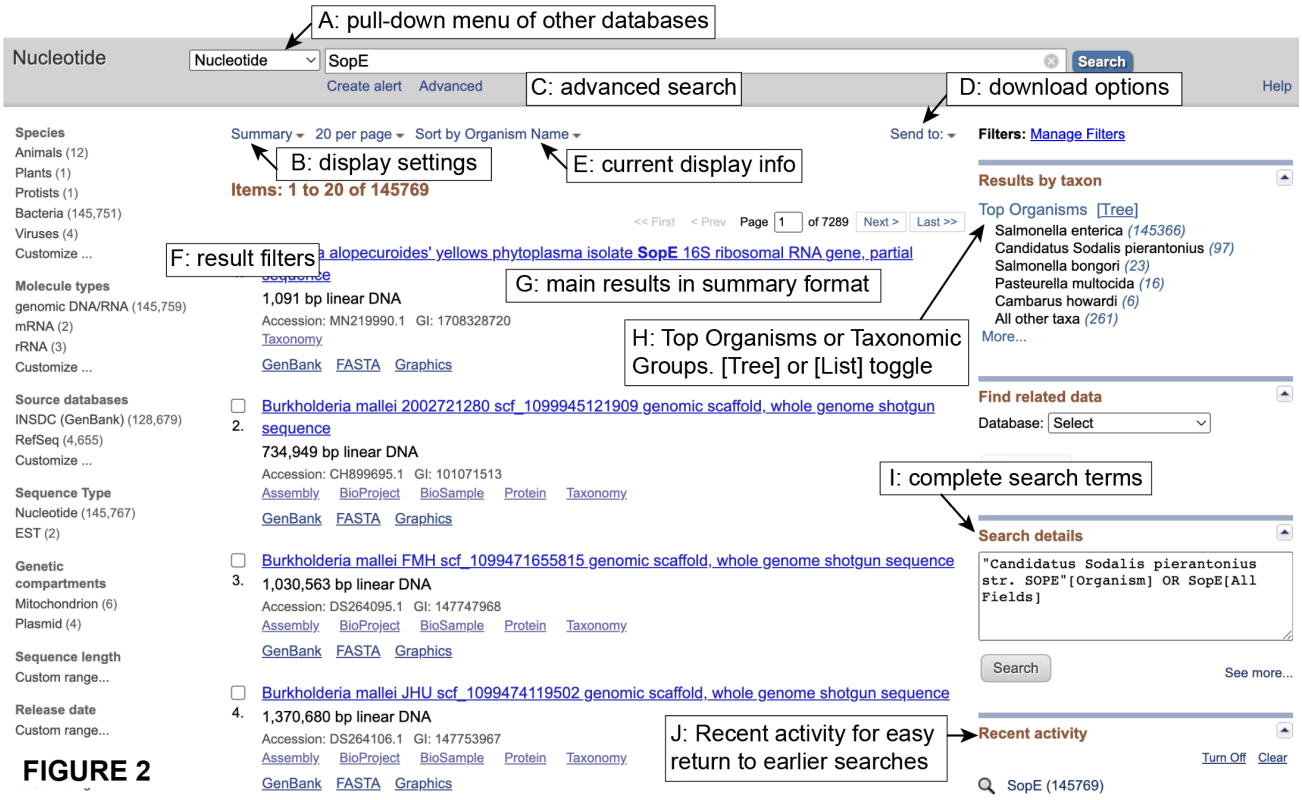


FIGURE 2

Observe the different types of records. Some hits are quite specific and will have SopE in their name. Other hits represent whole genomes/chromosomes, and contain sequences that are millions of base pairs long. These hits mention SopE somewhere in the record and thus come up in the search.

5. Explore some of the record links (perhaps in a new tab) to see how sequence data are presented. You can always return to your results list via **Recent activity** (**Figure 2J**).

1C. Advanced searches

Let's download *sopE* sequences of *Salmonella*.

1. On your results list from the **SopE** search against the Nucleotide database, filter the list to show only bacterial sequences.

Hint: see **Figure 2F or 2H,** or you can use the search bar. Your search terms would be similar to:

```
sopE AND (bacteria[Filter])
```

When results filter (Fig. 2F) is used, changes are not reflected on the search bar, but note (a) **Filters activated** on top of the Results list, and (b) your most recent search in the **Recent activity** panel (Fig. 2J).

2. Try limit the search to Salmonella using these terms:

```
sopE AND "bacteria"[Filter] AND Salmonella
```

3. Try toggling the **Top Organisms** between **Tree** and **List** (**Figure 2H**).

In tree format, the results are shown according to the established phylogenetic tree. e.g. g-proteobacteria (g for gamma) and b-proteobacteria (b for beta) groups, respectively.

Q6. *Burkholderia* are beta-proteobacteria. True or False?

You will notice that other bacterial records are still listed in the Taxonomic Groups panel (**Figure 2H**). *Why?* We did not specifically search for Salmonella in the field of source organism in the database. You can specify this in the search bar or use the **Advanced Search Builder** (**Figure 2C**) tool to do this.

4. Modify your search bar per below to restrict the search of "Salmonella" to the Organism (i.e. the source organism) field. Alternatively you can build this search using the Advanced Search Builder.

```
((SopE AND "bacteria"[Filter])) AND Salmonella[Organism]
```

5. The resulted list should contain only nucleotide sequence records from *Salmonella*. How many sequences do you get?

6. Have a look at the various options in **Display Settings** (**Figure 2B**) and how to sort the results using different attributes, format, and to display more records per page.

The default display format of the list is **Summary**. Note the sequence length and type of sequence (linear/circular) associated with each record. The accession and GI numbers are unique identifiers in NCBI.

The GenBank database contains a huge array of submitted sequences from many different groups, labs and technologies. As such, it can contain a lot of redundancy. In contrast, the RefSeq database is a curated database containing high-quality current knowledge of known genes, though it is less comprehensive than GenBank. RefSeq records consist of only genes and proteins from complete or draft genome sequences.

Sequences from other studies e.g. cloning and characterisation of single genes (that are not based on complete genomes) would not be available in RefSeq. In our case, we want to include all available, known *sopE* sequences, so we will use the GenBank (i.e. INSDC) records.

7. Filter the list to show only **GenBank** sequences (*Hint:* **Figure 2F**).

We are looking for sopE gene sequences, NOT entire genomes. Genomes of *Salmonella* are nearly 5 Mbp long. These genome sequence records have the word “genome” or “chromosome” in their DEFINITION field (note that the *Salmonella* genome is comprised of only a single chromosome). We want to exclude these.

8. Exclude these from your list by adding **`NOT genome[title] NOT chromosome[title]`** to your search.

Note that `[title]` refers to the **`DEFINITION`** field in GenBank records. You should end up with around 26 sequences.

1D. Downloading sequences in different formats

We want to download all sopE sequences from *Salmonella* from GenBank in FASTA format – the most commonly used flatfile format for storing sequences. **§ Supplementary Notes 4, 5**

Option 1. From the search results list, use the **Send to** option (**Figure 2D**) to save the sequences in a **File** (the destination) in FASTA format.

This approach is suitable for a small number of sequences and less likely prone to human error than the next option (copying/pasting of incomplete data); it saves all items in the search results.

Option 2. Display the search results list in **FASTA (text)** format, copy and paste the sequences into an editor (e.g. *Notepad* or *TextEdit*). Make sure you copied all sequences, not only the first 20 shown on the first page.

While you are here, have a look at the other formats available.

1. Back on your search results, look at the record of **`AF043239`** in GenBank format, one of the reported sopE sequence in *Salmonella*. Your screen should look like **Figure 3**.

FIGURE 3

The top part of the GenBank file (Header; **Figure 3B**) contains general information about the sequence. Below the header are the FEATURES (**Figure 3C**). Each feature has co-ordinates and descriptive information (i.e. the *annotations*). There are several protein sequences encoded in the nucleotide sequences (coding sequences, **CDS**); *sopE* contains both features of **gene** and **CDS**. For each CDS, other annotations include *note*, *product* and *translation*. **§ Supplementary Note 6**

Hint: If you can't find record AF043239 in your search results, try changing the display settings to show more **Items per page**.

Q7. How many CDS are found within the first 5000bp of AF043239?

In AF043239, positions corresponding to *sopE* CDS are **`complement(1560..2282)`**, indicating that *sopE* is encoded on the reverse strand in these positions. Other CDS encoded on the forward or *plus* strand do not have the **`complement`** notation.

2. Retrieve the *sopE* CDS from AF043239, in positions **`complement(1560..2282)`**.

Hint: Use **Change Region Shown** (**Figure 3D**) and **Customize View** (**Figure 3E**) to help you, and view the CDS in FASTA. You should see the *sopE* CDS (723 bp) that begins with **GTG** and ends with **TGA**.

1E. Integrating information from PubMed

PubMed is an up-to-date repository of all papers in Medline, the premier bibliographic database of the US National Library of Medicine. PubMed is integrated in NCBI, and is searchable using NCBI search. Some gene and protein database entries contain links to related published papers. **§ Supplementary Notes 7, 8**

1. On the AF043239 GenBank record, locate the published article that is associated with this record via **PubMed** link (this is a unique PubMed identifier).

The abstract of the paper is usually shown, along with figures. Full-text articles are linked in the top right where available. Sometimes, articles may appear to be paywalled if you are accessing PubMed from off-campus; logging into the UQ library website or the UQ VPN may allow you access to such papers.

2. Have a look at the **Figures** from this publication. These figures give us an indication that a great deal of molecular and phenotypic characterisation of the *sopE* gene was carried out.

3. On the PubMed record, access the SopE protein related to this publication via the **Protein** link in the **Related information** section (near the bottom of the page).

This should bring you to the NCBI GenPept record for O52623. This is a **UniProtKB / Swiss-Prot** record. The Swiss-Prot database (which falls under the UniProtKB database umbrella) is a manually curated database of protein sequences which includes a great deal of information compiled in its annotations.

4. View the **O52623** GenPept record. Have a look at the COMMENT section.

Hint: If a protein sequence you are looking for has a Swiss-Prot record, this is a good starting place when trying to determine its function – a lot of existing information is included here and it may save you a lot of time and effort.

The following two questions test your ability to use NCBI's search:

Q8. Find the following information based on GenBank nucleotide record **AY150213**:

(a) first three nucleotides in the CDS of tetA protein,
(b) the last three nucleotides in the CDS of tetR protein,
(c) the PubMed identifier of the paper related to this record,
(d) the journal in which the paper was published, and
(e) the UniProtKB/Swiss-Prot identifier for the protein related to the paper.

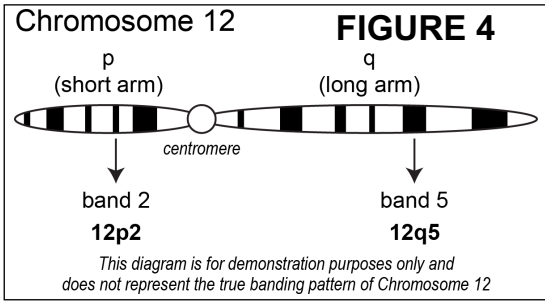
Q9. What is the Accession number of the RefSeq record for the pHCM2 plasmid sequence from *Salmonella enterica* subsp. *enterica* str. CT18? How long is the sequence?

PART 2. Browsing genome maps using Ensembl

In human genetics, cytogenetic maps are based on the differential banding pattern observed in condensed chromosomes after staining. **§ Supplementary Note 9**

There are several browsers for genome maps of human and other organisms, e.g. **Genome Data Viewer** at NCBI (https://www.ncbi.nlm.nih.gov/genome/gdv/), **UCSC browser** (https://genome.ucsc.edu/) and **Ensembl** (https://ensembl.org/). Here you will learn how to view, compare, and interpret different genome maps using **Ensembl Genome Browser**, and to use these maps to access additional genome and annotation data.

Did you know? Australia is an associate member of the European Molecular Biology Laboratory (EMBL). For more info, see https://www.emblaustralia.org/.



In humans, a gene with cytogenetic coordinates 17q12 is located at the long arm (q) of chromosome 17 at band position 12. A gene with coordinates 5p14 is located at the short arm (p) of chromosome 5 at band position 14. The cytogenetic band numbers increase as the distance from the centromere increases, e.g. 5p1 is closer to the centromere of chromosome 5 than 5p14 is. Other nomenclature systems are used in other species. **§ Supplementary Note 9**

Q10. Cytogenetic coordinates for a human gene *UBE2D2* is determined at 5q31.2. Another gene *GCNT4* is located at 5q13. Which of these two genes are closer to the centromere of chromosome 5?

2A. Accessing the Ensembl Genome Browser

1. Retrieve the `Prac1_sequence.fasta.txt` file from Blackboard at ***Core Practicals > Practical 1***. This is your unknown sequence.

Open the file and check to make sure it looks as expected – sometimes direct downloads from Blackboard include HTML in .txt files.

2. On Ensembl (www.ensembl.org), go to **BLAST/BLAT** tool (link on top of page), use the unknown sequence as query (i.e. copy and paste the sequence into *Sequence data* box, or upload the file).

We will re-visit the BLAST tool in more detail in Practical 2. For this week, know that BLAST and BLAT are both methods of comparing a DNA or protein query sequence to other sequences in a database, to find similar sequences. In this case, we will use BLAT to find the identity of our unknown sequence. BLAST and BLAT function similarly, but BLAT searches for exact or near-exact hits (making it ideal for rapid queries of very similar sequences) while BLAST is more flexible and can identify more distant matches.

3. Run **BLAT** against **Human (Homo_sapiens)** Genomic Sequence (DNA) database.

Make sure the *Sequence data*, *Search against* and *Search tool* options are set correctly.

4. The progress of the run is shown on the auto-refreshing screen. When the run is finished, a green **Done** box will appear, next to a **[View Results]** link. Have a look at the Results.

You should see a list showing high-scoring pairs (HSPs); the longest HSP on top. Each **[Alignment]** link shows the aligned region in detail. A cytogenetic map showing HSP distribution is shown below the table.

Q11. Based on your BLAT results:

(a) How many hits (HSPs) do you get in the BLAT analysis?

(b) In which human chromosome are these hits found?

(c) What is the length of the longest HSP?

(d) To which chromosome positions does the longest HSP match (i.e. the start and end positions of the chromosome that correspond to the HSP match)?

(e) Which positions of the query sequence are identical to positions 102533845-102533885 of chromosome 4?

2B. Examining genomic regions on a map

1. Click on the **Genomic Location** for the top hit (the longest HSP). This brings up a location-based display corresponding to the matching sequence in the genome. Your screen should be similar to **Figure 5**; this view is also referred to as the ContigView.

Have a look at the chromosome ideogram at the top of the page showing the different band regions of the p and q arms of Chromosome 4. The dark and white banding pattern indicates different isochores of the cytogenetic map of the chromosome (chromosome bands). The position of our match is marked in a red rectangle. Below this is a blow-up view of the region corresponding to the red rectangle – your match falls within the NFKB1 gene, as indicated by a red vertical line in the centre of the panel. The genes are depicted as coloured rectangles with labels indicating the gene names. The direction of the arrow next to each gene name indicates the orientation of each gene on the chromosome.

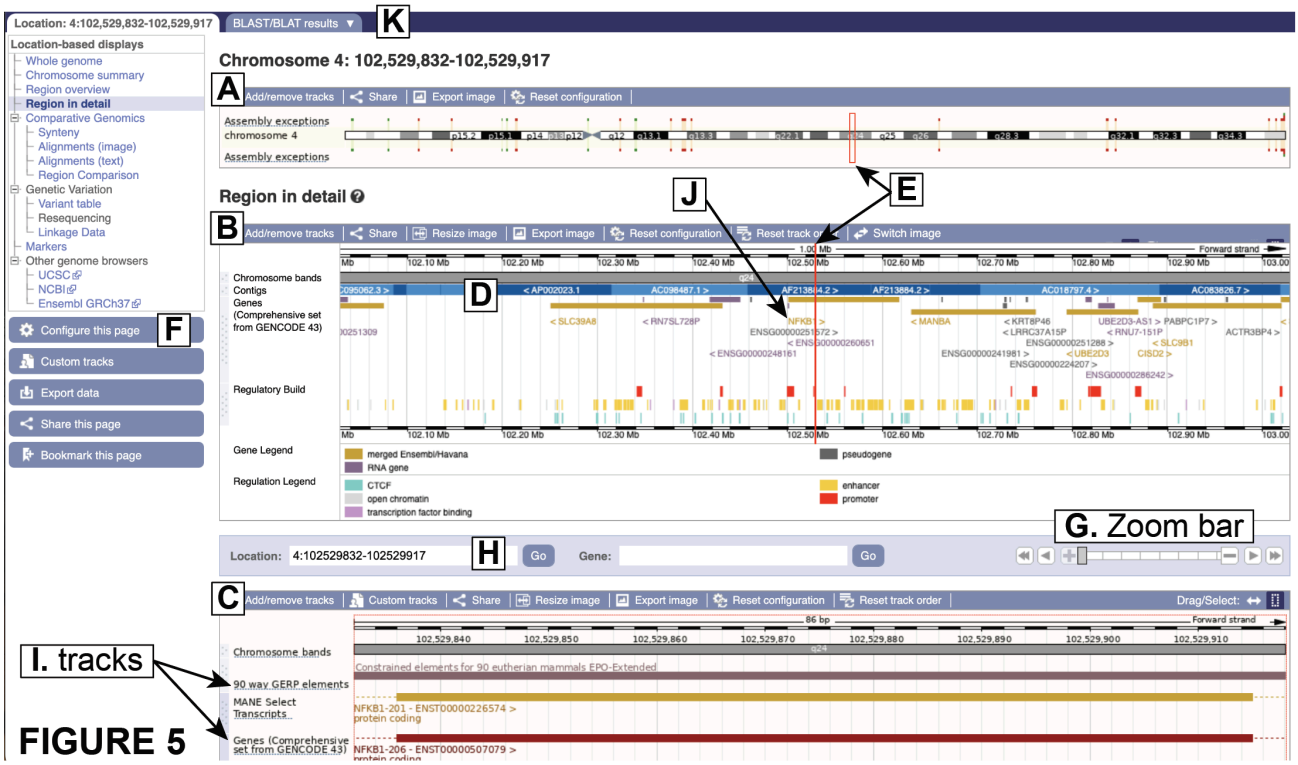


FIGURE 5

ContigView (map) in Ensembl

The upper panel (Chromosome) shows an **ideogram** of the displayed chromosome (4, in this case; **Figure 5A**), indicating the cytogenetic bands. As you can see, the NFKB1 gene is located at band q24 of the chromosome. The next panels display the 1 Mb region (**Figure 5B**) and HSP region (**Figure 5C**), respectively. It includes an overview of the genes and markers in the region.

A scale bar in each panel illustrates the physical map coordinates for the region. The band in alternating shades of blue (**Figure 5D**) show individual contig sequences that form the genome sequence assembly.

The bottom panel is very rich in features and can be customised. Note that this is on a different scale than the Overview panel, and it corresponds to only the region highlighted by the red rectangle above (**Figure 5E**), i.e. only one exon is shown in **Figure 5C**, whereas several whole genes are shown in **Figure 5B**. The features shown in **Figure 5C** are called tracks.

Tracks can be turned off or on by using the **configure this page** link (**Figure 5F**). Turning off unwanted features and functions will make the Web pages download faster and make it easier to see the features of interest. After you checked/unchecked the track selection, click on the tick mark (top right corner) to save and exit; this will update the view. **§ Supplementary Note 10**

2. Zoom in to show the whole NFKB1 gene. You can drag the red box in "Region in detail" such that the window would cover the entire gene.

3. Look at the track labelled ***Genes (Comprehensive set …)***. Click on **Comprehensive set...** to open a pop-up box and hover over the info icon to see an explanation of this track.

This section shows all the transcripts that have been sequenced for this region. For each transcript shown, a thin line (**introns**) connects small vertical boxes (**exons**).

4. There should be track named ***90 way GERP elements*** towards the top of the third panel. If you do not see the track, you can load the track by using **Configure this page** (**Figure 5F**) > **Comparative genomics** section > **Conservation regions** > activate the **Constrained elements for 90 eutherian mammals EPO-Extended** track.

These databases are updated very frequently. If you can't find this precise option, look for "Constrained elements for XXX eutherian mammals EPO-Extended" – the database may have been updated to include additional species.

The **Constrained elements** tracks show regions in the genome that are evolutionarily conserved in genomes of other species, and therefore more likely to be functional. This is because functional DNA is under stronger evolutionary pressure to remain the same, whereas non-functional DNA is less constrained and is thus expected to change a great deal over time. This evolutionary constraint is measured using the GERP (Genomic Evolutionary Rate Profiling) score. Here, the tracks show that the highly conserved regions generally correspond to exons. In the NFKB1 example, there are also several other highly conserved regions (or constrained elements) that do not correspond to exons, suggesting they may be undiscovered exons, non-coding genes or regulatory elements.

Q12. There are no constraint elements between exons 1 and 2 of NFKB1-201 transcript. True or False?

2C. Viewing synteny between the human and mouse genomes

1. Chromosomal regions that are homologous between two species - that is, they are derived from a single ancestral genomic region - are said to be syntenic. Synteny information can be viewed in the Ensembl browser via the menu panel on the right: On current ContigView (showing whole region of NFKB1), view synteny information for Chromosome 4 via the **Location-based displays** menu panel on the left: **Comparative Genomics > Synteny**. By default the comparison is with the mouse (*Mus musculus*) genome.

Synteny information is shown for the whole chromosome, and the location of the gene of interest in both human and mouse is shown with a red rectangle.

Q13. Human chromosome 4 contains regions that are syntenic to regions in which four chromosomes in mouse?

2. At the bottom of the page there is some information about the mouse homologue of NFKB1, which is called **Nfkb1**. Open the location link in a new tab (3:135,290,416- 135,397,308) and you will be taken to a ContigView page for mouse gene.

Q14. Which chromosome is the NFKB1 orthologue (Nfkb1) located in mouse, and on which cytogenetic band (isochore)?

3. Compare the ContigViews of mouse Nfkb1 and human NFKB1 regions.

Note that the mouse transcripts are in the opposite orientation due to an **inversion** between human and mouse genomes in this region. The exon architecture for NFKB1 is also different in human and mouse. These genes have evolved from a common ancestor. Although the CDS remain relatively conserved, non-coding elements are not.

4. Bring up the **GeneView** for mouse Nfkb1. It should be in the tab next to the ContigView of mouse, or you may simply search for it on Ensembl.

5. Retrieve all orthologues for this gene via the menu on the left panel: **Comparative genomics** > **Orthologues**. Scroll down to the human (*Homo sapiens*) Nfkb1 orthologue and view the pairwise **protein** sequence alignment via the **View Sequence Alignments** link.

Q15. What is the percent sequence identity between Nfkb1 mouse protein and its ortholog in human? Which protein is longer?

6. GeneView also provides detailed information about transcript variants. Click **show transcript table** so see the list (it might be hidden). Note the links to CCDS (Consensus CDS database) and UniProt records.

Q16. How many CCDS records are associated with Nfkb1-201?

Concluding remarks

At this point you should be familiar with the resources available at NCBI and know how to access the specific, relevant information you need. Specifically, you should know how to use Boolean operators in advanced search, download the relevant data from NCBI, and search for publications indexed in PubMed. Many of these skills apply to other biological resources online as well.

You should be able to use the Ensembl Genome Browser to search for the genes you are interested in, have a basic understanding of ContigView and GeneView, and how to customise the browser to show the information you are interested in. You should also be familiar with basic concepts of genome mapping, e.g. cytogenetic maps, and how to view synteny between two genomes under comparison.

That might seem like a lot, but you will soon see how these skills are handy as you proceed with the course.

ANSWER KEY

Q1. DNA DataBank of Japan (DDBJ), the European Molecular Biology Laboratory (EMBL), and GenBank at NCBI

Q2. False (RefSeq contains non-redundant sequences)

Q3. Sequence Read Archive (SRA)

Q4. **AND**

Q5. (a) 19 (b) 143,650
[These numbers could be slightly different, as they will change when the database is updated with relevant entries]

Q6. True

Q7. 7

Q8. (a) ATG (note the coding on the reverse complement strand)
(b) TAA
(c) 15280224
(d) Genetics
(e) Q8Z3V1.1

Q9. NC_003385.1; 106,516 bp

A possible search construct:

```
Salmonella enterica[organism] AND
pHCM2[title]
```

Q10. *GCNT4*

Q11. (a) 4 (b) 4 (c) 79
(d) 102529836-102529914 (e) 119-159

Q12. False

Q13. Chromosomes 3, 5, 6 and 8

Q14. Chromosome 3, long arm q

Q15. 85-86%; the ortholog in mouse (Nfkb1) is longer

Q16. 1

Practical 2: Sequence Alignment & Similarity Searching

Sequence alignments are key to allow two or more sequences to be compared, for example to predict evolutionary or functional relatedness between sequences. In this practical we will learn about the methods that have been developed to align pairs of sequences, including dot-plot, dynamic programming and BLAST. **§ Supplementary Note 1**

Learning objectives:

A. To align sequences using dot-plot
B. To understand sequence alignments using dynamic programming
C. To understand the differences between local and global alignments
D. To understand key parameters in a sequence alignment
E. To search databases for similar sequences using BLAST

Additional references:

- Textbook – Zvelebil NJ & Baum JO (2008) *Understanding Bioinformatics,* Garland Science:
  - *Chapter 4* Producing and Analyzing Sequence Alignments (p72-102)
  - *Chapter 5* Pairwise Sequence Alignment and Database Searching (p117-139)
- See also **Practical 2 Supplementary Notes** and **Videos** on Blackboard

A. Align sequences using dot-plot

In a **dot-plot** (or **dot matrix**), one sequence is placed horizontally on top, another vertically down the side, and any cell with a character match between the sequences is marked. § Supplementary Note 2

↘	C	A	T	G	A
A		■			■
A		■			■
C	↘				
A		■			■
T			■		
G				■	

FIGURE 1

1. Consider **Figure 1**, comparing two sequences: **CATGA** and **AACATG**. Trace diagonally (upper-left to lower-right), note the longest match is 4 characters: **CATG**.

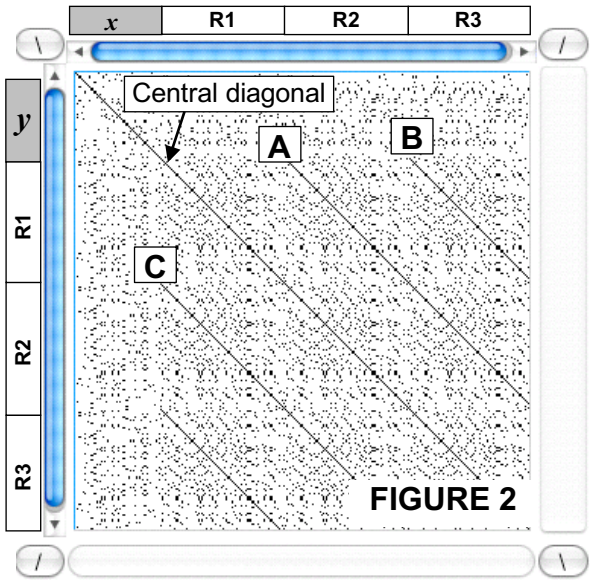


FIGURE 2

Alignment of much longer sequences is shown in the dot-plot in **Figure 2.** Here we can also identify tandem (*one after another*) repeats in a sequence.

Line **A**: R1 & R2 on y matching R2 & R3 on x

Line **B**: R1 on y matching R3 on x.

Line **C**: R1 & R2 on x matching R2 & R3 on y.

The lines below the central diagonal are the mirror image of lines **A** and **B** above. This pattern of diagonals is typical of tandem repeat sequences.

2. Consider the sequence **ATGCATCATCATTTA**: it has 3 identical tandem repeats **CAT**. Try completing the dot-matrix plot below for the grey box region containing the 3 repeats.

The length of the shortest diagonal in the grey block (3 nucleotides in line **1**) is the length of the repeat. The total number of repeats is the number of diagonal lines above the central diagonal in the upper-right half of area **A** PLUS one (*in this case, there are 2 lines + 1 = 3 repeats*).

Note the long rectangular box pattern (**Fig. 3B**) near the end of the sequence, as caused by **TTT**, a ***homopolymeric tract*** (a tract of a single character).

Q1. Sequence ***M*** has 7 tandem 10-nucleotide repeats. Excluding the central diagonal, how many lines would one expect to find above the central diagonal line (area **A** in **Figure 3**) in a dot-plot comparing ***M*** to itself?

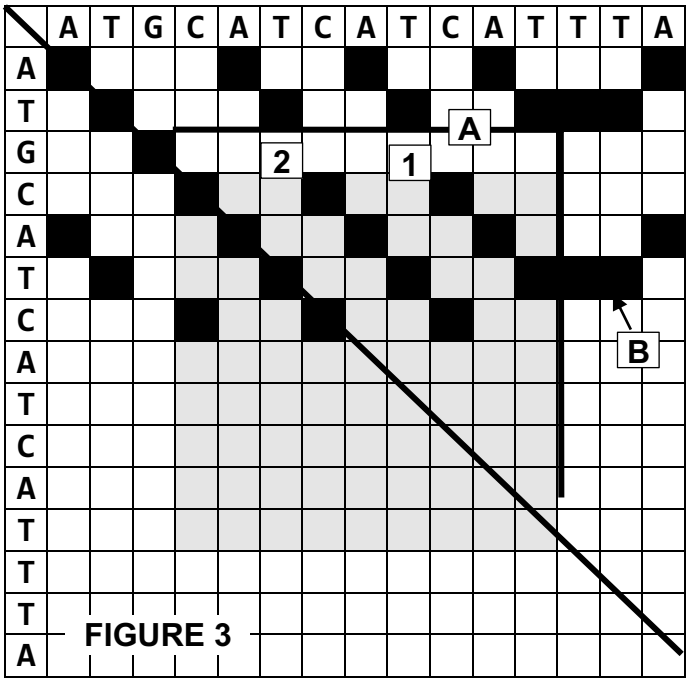
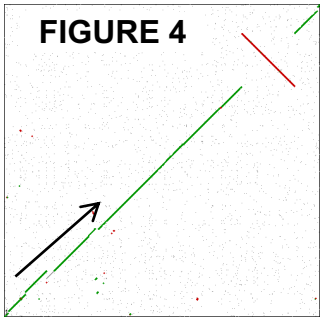


FIGURE 3

3. Consider the dot-plot (**Figure 4**) that compares between two bacterial genomes. *In this case, the sequences are compared from bottom up following the arrow.*

Note the few (small) differences indicated by the broken lines – these occur when there is an insertion or deletion in one of the genome sequences. There is also a rearrangement (the red line) in one genome relative to the other.



Q2. Consider the dot-plot in **Figure 4**. Is the red line indicative of a translocation, duplication or inversion?

B. Dynamic programming in sequence alignment

Dynamic programming is a common approach in comparing similarity between two sequences. The Needleman-Wunsch (1970) algorithm is the first of such methods. An alignment with the maximal score is the **optimal** solution. **§ Supplementary Note 3**

↘	M	A	D	A	M
M	M M				
A		A A			
D			D D		
M			- M		
A				A A	
N					M N

FIGURE 5

Consider two protein sequences **MADAM** and **MADMAN**. In dynamic programming, these are placed in a matrix – just like a dot-plot. The optimal alignment can be seen as highest scoring diagonal path through the matrix. Where a gap is needed to maintain the highest scoring path, a move to the right or down is required.

The matrix in **Figure 5** produces

```
MAD-AM
MADMAN
```

An alternative alignment is shown on the right, but this requires the insertion of 3 gaps, rather than 1. Gaps are usually weighted negatively, so the 2nd alignment will score less than the 1st. Therefore, this alternative alignment is *not* optimal.

```
MAD-A-M
MADMAN-
```

Dynamic programming is a two-phase process. We will use the Needleman-Wunsch (NW) algorithm as an example.

Phase 1: Analysis. Fill the matrix with scores and back-pointers
Phase 2: Evaluation. Find the highest scoring diagonal by *tracing back* the high-scoring route through the matrix using back-pointers

X → Y ↓	*gap*	T	A	G	C
gap	← 0	← -3	← -6	← -9	← -12
T	↑ -3				
A	↑ -6				
C	↑ -9				

FIGURE 6. Needleman-Wunsch example.

Phase 1

Sequences: **SeqX** = **TAGC**, **SeqY** = **TAC**.
Scoring scheme: match (**+3**), mismatch (**0**), gap (**−3**).

(a) First we draw a matrix like we did in the dot-plot. **SeqX** goes on the *x*-axis, **SeqY** goes on the *y*-axis. Then we set the top and left sides of the matrix to multiples of the gap penalty, (**−3** in this case), and **initialise their back-pointers** (the arrows indicating where the accumulative score was obtained). The scores are accumulative from left to right and from top to bottom at this stage. The role of the back-pointers will become clearer once we move from Phase 1 to Phase 2.

(b) To calculate score for box labelled **#**, first we assess the scores for each box (i) **above** (−3), (ii) **above-left** (0), and (iii) **left** (−3) of it. If we were tracing from the box:

X → Y ↓	*gap*	T	A	G	C
gap	← 0	← -3	← -6	← -9	← -12
T	↑ -3	→ #			
A	↑ -6				
C	↑ -9				

(i) above, a gap is introduced in **seqX** (−3).

(ii) on the **left**, a gap is introduced in **seqY** (−3).

(iii) on **above-left**, we are comparing the character pair for box **#**. Here, it is a match of **T** to a **T** (3).

Consider the sum of each score in the considered boxes with the corresponding score in condition (i.e. the direction of) **(i)**, **(ii)** and **(iii)**:

```
Above (gap in seqX): (-3)+(-3) = -6
Left (gap in seqY) : (-3)+(-3) = -6
Above-left (match) : (0) + (3) = 3
```

Choose the maximum of these scores. This is our score for box **#**. We will record this score in box **#**, and add a **back-pointer** that traces back to the winning box (see below).

X → Y ↓	*gap*	T	A	G	C
gap	← 0	← -3	← -6	← -9	← -12
T	↑ -3	↖ 3	@		
A	↑ -6				
C	↑ -9				

(c) We then calculate the scores for the remaining boxes, now **@**. We repeat the process from **(b)**:

```
Above (gap in seqX)   :(-6)+(-3) = -9
Left (gap in seqY)    : (3)+(-3) = 0
Above-left (mismatch):(-3)+ (0) = -3
```

The character pair in box **@** is **A** in **SeqX** and **T** in **SeqY**, hence it is a mismatch (0).

(d) The maximum of the three scores from **(c)** is entered in box **@** and a back-pointer is recorded. This process then continues from left to right, row by row, until the matrix is filled with scores and back-pointers. Once the matrix is filled, the highest scoring path is *traced back* through the matrix by following the arrows, starting at the bottom right box (**Phase 2**).

X → Y ↓	*gap*	T	A	G	C
gap	← 0	← -3	← -6	← -9	← -12
T	↑ -3	↖ 3	← 0	← -3	← -6
A	↑ -6	↑ 0	?	← 3	← 0
C	↑ -9	↑ -3	↑ 3	↖ 6	↖ 6

Using the optimal trace-back path, the alignment can be built step-by-step, working from the end of the alignment back to the start i.e. **C:C**, **G:***gap*, **A:A**, **T:T**. This representation is widely used for alignments, with "**:**" denotes identical matches, "**.**" denotes similar matches (conserved substitutions – used in protein alignments), a blank denotes columns with a gap or mismatch.

```
seqX TAGC
     :: :
seqY TA-C
```

Tip: This is a simple example, but if you're having trouble getting the gaps in the right place when writing out bigger alignments, remember that the arrow points to the sequence with the gap i.e. here there is a gap in seqY, and the arrow points to the seqY side.

Q3. Consider the box labelled "?" in step (d) above.
(a) What is the score in the box?
(b) Which direction is the back-pointer pointing in: left, diagonal, or up?

NW algorithm is a **global** alignment algorithm. Smith and Waterman (1981) modified the NW algorithm to find the locally matched regions between two sequences (i.e. a **local** alignment algorithm). In this algorithm: (a) the scoring system includes negative scores for mismatches, (b) negative scores are set to zero, and (c) during Phase 2 (evaluation), begin at the square with the highest score, and trace that alignment until a zero is reached. **§ Supplementary Note 4**

X → Y ↓	*gap*	T	A	G	C
gap	0	← 0	← 0	← 0	← 0
A	↑ 0	↖ 0	↖ 3	← 0	0
G	↑ 0	↖ 0	↑ 0	↖ 6	← 3
T	↑ 0	↖ 3	0	↑ 3	↖ 5

Figure 7. This time our scoring system is: **match = 3, mismatch = −1, gap = −3**. Once again we populate the matrix using the dynamic programming calculation; only this time, the first row and column are populated with **zeros** rather than negative numbers. Trace-back begins with the **highest scoring position** in the matrix, and **proceeds until a zero** is reached. Therefore, the best local alignment is:

```
seqX AG
     ::
seqY AG
```

C. Local versus global sequence alignment

As we saw above, sequence alignments can be one of two types, **local** or **global.** The selection of alignment type will depend on your analysis goals. Local alignments aim to find high-similarity match regions between two sequences, whereas global alignments attempt to align the entire length of the sequences, end-to-end.

Here we will observe these effects of local alignments vs global alignments, by testing both methods on the same set of sequences.

1. Retrieve **`SeqA.fasta.txt`** and **`SeqB.fasta.txt`** from Blackboard site (**Practical 2**).

2. First, run a **local** alignment. Copy and paste each of these protein sequences as *First* and *Second* sequence respectively on the EBI LALIGN server (https://www.ebi.ac.uk/Tools/psa/lalign/). Do not alter any parameters.

3. Submit the job to run a **lalign local** alignment on these sequences and view the results (all high-scoring local alignments will be shown, not just the optimal alignment).

Points to note:
(i) The best alignment stretches from the start of the sequences, and does not include the C-terminal regions (i.e. note that the aligned region is shorter than the length of the sequences).
(ii) There is a 13 amino acid gap in the N-terminal region of **seqA**.

4. Now, perform a **global** alignment between the two sequences, using EMBOSS Needle (https://www.ebi.ac.uk/Tools/psa/emboss_needle/) at the default parameters. Copy-and-paste **seqA** and **seqB** into appropriate boxes (as in **step 2**). Do not alter any parameters.

Notice that the sequences match perfectly (aside from the deletion near the beginning) until about the last 100 amino acids. Note the alignment score. The quality of the matches appears to degenerate, but the alignment was forced right through the end. Do you think there is any significant similarity between the C-terminal region of SeqA and SeqB?

5. We will test this using **`SeqA_Cterm100.fasta.txt`** and **`SeqB_Cterm100.fasta.txt`**, which contain only the C-terminal region of each protein. Perform another global alignment at default settings. Note the alignment score.

6. How do these results differ from random? Using **`SeqA_Cterm100.fasta.txt`** again and **`SeqB_Cterm100random.fasta.txt`** (the same amino acids from **`SeqB_Cterm100.fasta.txt`** but shuffled into randomised order), perform global alignment (EMBOSS Needle) at default settings. Note the alignment score.

Do you still think there is any significant similarity between the C-terminal region of SeqA and SeqB?

This exercise demonstrates how sequence alignments can be misleading. You have performed global alignment directly by comparing only the C-terminal regions (**step 5**), and against a randomised sequence (**step 6**). Which one has a higher score?

Global alignment aligns all regions in both sequences in order to find the best alignment. If you use two sequences that are not related, it will still find the optimal alignment, but this alignment will be **biologically meaningless.** It is important to use the appropriate tools to address your specific research problem.

Q4. Which method (local or global) would be the most suitable for:
A. aligning sequences of a similar length that you already know are homologous.
B. aligning gene sequences to genomic sequences (that include introns).

D. Key parameters in a sequence alignment

The choice of a scoring system (including scores for matches, mismatches, substitutions, insertions and deletions) would affect any sequence alignment. In the previous examples we have scored a gap as -3, a gap of two positions as -6 and gap of three positions as -9. Is this reasonable?

Think about this in another way: **SeqX** and **SeqY** arose from a single ancestral sequence, SeqA underwent a 9-nucleotide deletion and SeqB underwent a 3-nucleotide deletion. In this hypothetical example, in both cases it was a single evolutionary event that caused the gaps to appear (i.e. instead of 9 events of single-nucleotide deletion in SeqA). Although it is true that a deletion of 9 nucleotides might be less likely than a deletion of 3 nucleotides, it is not 3 times less likely.

The biological reality of gap creation led to the development of the **affine gap penalty**. This gap penalty score (**`G`**) is a linear function of gap length (i.e. the length of inserted/deleted region, **`L`**) relative to gap opening (**`O`**) and gap extension (**`E`**):

$$G = O + E \times (L-1)$$

A gap penalty **score** is commonly denoted as a **negative value**, i.e. the values of **`G`**, **`0`**, and **`E`** are usually denoted as negative. *The smaller the score, the greater the penalty.* In most cases, the gap extension penalty is much less than the gap opening penalty – making it harder for gaps to be opened in an alignment, but once opened they can be extended substantially without penalising the overall alignment score too much.

Q5. Consider `SeqX (AAATTTAAA)` and `SeqY (AAAAAA)`, and this scoring scheme: match (+3), mismatch (−2), gap opening penalty (−3) and gap-extension penalty (−1).

A. What is the score of the optimal alignment between **`SeqX`** and **`SeqY`**?

B. What would be the optimal alignment between **`SeqX`** and **`SeqY`** if all other parameters in the scoring scheme stayed the same, but the gap-open penalty is now −20?

A **substitution matrix** is a matrix containing the information on the frequency of mutation of one residue to another. **Figure 8** shows an amino acid substitution matrix, with a *log-odds score* assigned to each possible pair of amino acids. A **log-odds score** represents the likelihood that two different amino acids are aligned, divided by the likelihood that they are aligned simply by chance (i.e. purely due to their observed frequencies in the sequences – this is the null model). **§ Supplementary Note 5**

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	2																			
R	-2	6																		
N	0	0	2																	
D	0	-1	2	4																
C	-2	-4	-4	-5	4															
Q	0	1	1	2	-5	4														
E	0	-1	1	3	-5	2	4													
G	1	-3	0	1	-3	-1	0	5												
H	-1	2	2	1	-3	3	1	-2	6											
I	-1	-2	-2	-2	-2	-2	-2	-3	-2	5										
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	2	6									
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5								
M	-1	0	-2	-3	-5	-1	-2	-3	-2	2	4	0	6							
F	-4	-4	-4	-6	-4	-5	-5	-5	-2	1	2	-5	0	9						
P	1	0	-1	-1	-3	0	-1	-1	0	-2	-3	-1	-2	-5	6					
S	1	0	1	0	0	-1	0	1	-1	-1	-3	0	-2	-3	1	3				
T	1	-1	0	0	-2	-1	0	0	-1	0	-2	0	-1	-2	0	1	3			
W	-6	2	-4	-7	-8	-5	-7	-7	-3	-5	-2	-3	-4	0	-6	-2	-5	17		
Y	-3	-4	-2	-4	0	-4	-4	-5	0	-1	-1	-4	-2	7	-5	-3	-3	0	10	
V	0	-2	-2	-2	-2	-2	-2	-1	-2	4	2	-2	2	-1	-1	-1	0	-6	-2	4

FIGURE 8

Identical matches: positive log-odds score, depends on the rarity of the amino acids e.g. Tryptophan-Tryptophan match scores much higher than Glycine-Glycine match.

Conservative matches: lower score that are also positive (substitutions that are more likely than by chance alone – i.e. amino acids that are physically or chemically similar).

Non-conservative matches: negative log-odds score (substitutions that are less likely than chance alone – i.e. amino acids that are physically or chemically very different).

Neutral matches: zero log-odds score (substitution would be expected by chance).

Log-odds scores are expressed in the **base of 2** (*binary logarithm*). The sum of log-odds scores in a pairwise alignment results in the total alignment score (also known as the *bit* score).

Q6. Referring to **Figure 8**, what is bit score (sum of log-odds scores) of aligning **`RAT`** to **`RAM`** with no gaps?

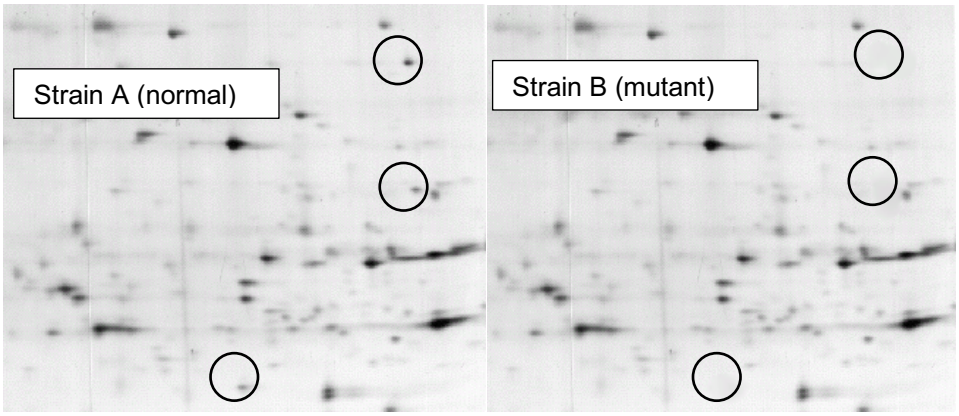
E. Searching databases for similar sequences

Similarity searches against sequence databases are a key task in bioinformatics. For example, you may want to find the identity of an unknown sequence, or find related sequences in a different species. The most widely used tool is the Basic Local Alignment Search Tool (**BLAST**). In BLAST, the **expect** or **expectation value** (**E-value**) roughly translates to the number of alignments with a score equivalent to or better than the observed bit score, which one would find simply by chance in the database search. **The smaller the E-value, the more significant the hit is.** § Supplementary Notes 6 & 7

Q7. All other things being equal, a sequence similarity match with E-value of 0.01 is less likely to be due to chance than a match with an E-value of 1. True or False?

Case Study 1

The aim of this experiment is to identify new effector proteins produced by *Escherichia coli* O157:H7 using proteomics. Effector proteins are only secreted by the type III secretion apparatus (the molecular needle), whereas many other non-effector proteins are secreted by other means. We want to compare the protein profile normally secreted by *E. coli* (strain A) with the protein profile of an *E. coli* mutant strain (strain B). The mutant strain cannot secrete effectors because its molecular needle has been disabled. By identifying proteins that are secreted by strain A but not strain B, we may be able to identify new effectors. Protein profiles can be compared using 2D electrophoresis:



Circles show protein spots that are missing from the mutant strain but present in the normal strain, suggesting that they are proteins that are normally secreted by the type III secretion needle. i.e. the only difference between the two strains is that Strain B lacks the needle because of a mutation.

As you can see, three spots on the protein gels are missing in Strain B but present in Strain A. We have excised one of these spots from the gel, sent it for N-terminal amino acid sequencing and a few days later we were told that the N-terminal sequence is **`MLSPSSINLGCS`**.

How do we find out what this protein is? Let's use BLAST to find out.

1. Navigate to the NCBI BLAST page (https://blast.ncbi.nlm.nih.gov/). Note the different BLAST programs available, depending on the types of query sequence and database.

2. Go to **Protein BLAST**. Use the sequence **`MLSPSSINLGCS`** as query, against the **Standard Databases** (nr etc.), with the *Algorithm* **blastp (protein-protein BLAST)**.

3. As our experiment is about *E. coli* O157:H7, we can simplify our search by searching only for sequences from this organism. In the **Organism** box enter and select ***Escherichia coli* O157:H7 (taxid:83334)**.

4. Check **Show results in a new window** and run **BLAST**.

You should see a message telling you that the search parameters were adjusted to search for a short input sequence – so BLAST is altering the parameters to achieve the best results. When the results come up you should see a BLAST results page with tabs directing to four main parts.

Page header. This contains basic information about your search and links to: (a) **Edit** the search, (b) **Save Search Strategies**, and (c) **Download**. Under the header are four tabs which link to other reports, i.e. **Descriptions**, **Graphic Summary**, **Alignments** and **Taxonomy**. There is also a link (top right) to the **BLAST help videos** on YouTube.

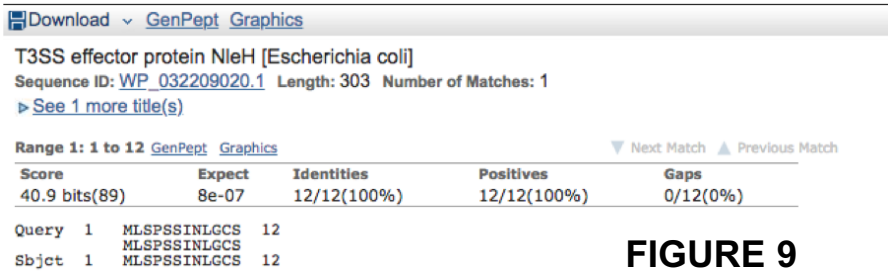
Descriptions. A list of *Description* of matches with links to the alignments, the associated scores, E-values, percentage identities, and the GenBank accession numbers (linking corresponding GenBank records).

Graphic Summary. Select some/all of the matches (**BLAST Hits**) and click over to the **Graphic Summary tab**. The distribution of chosen hits is shown by horizontal bars in a graphic, colour coded according to alignment scores. In this case, the whole query sequence is covered by the BLAST hits, but this view would show large gaps between aligned regions as breaks in the coloured lines.

Alignments. Pairwise alignments of each BLAST hit arranged in order of significance.

Taxonomy. Allows the user to browse the taxonomy of hits, and limit searches to particular species.

5. Click on the top hit link in *Descriptions*. You will be brought to the alignment for this match, similar to **Figure 9**:


Download GenPept Graphics

T3SS effector protein NleH [Escherichia coli]

Sequence ID: WP_032209020.1 Length: 303 Number of Matches: 1

See 1 more title(s)

Range 1: 1 to 12 GenPept Graphics Next Match Previous Match

Score	Expect	Identities	Positives	Gaps
40.9 bits(89)	8e-07	12/12(100%)	12/12(100%)	0/12(0%)

```
Query  1   MLSPSSINLGCS  12
           MLSPSSINLGCS
Sbjct  1   MLSPSSINLGCS  12
```

FIGURE 9

Each match will have an alignment similar to this, often with more than one "description" line for each matched protein sequence. Note the **E-value** and **bit score**. This example illustrates that you should be wary about using a strict E-value cut-off that is very conservative (e.g. 1e-10) as you may miss many true positives (such as this one) – although this match is 100% identical, the E-value is 8e-07 (8×10^{-7}) and would be lost with too strict a threshold. As you scroll down and view some of the other alignments, note that they get progressively worse as you go down the page i.e. the scores get lower and the E-value gets larger. E-values are dependent on database size, so can change as the database expands.

6. Click **Edit Search** on top of the page. Delete **Escherichia coli O157:H7 (taxid:83334)** from the **Organism** box (**step 3**). We will search the entire **nr database**.

Have Have a look at the **Algorithm parameters** (below the BLAST button, click the + to expand). When you are more experienced, it is possible (and often desirable) to over-ride these settings manually. Click on the "?" icon beside each parameter to learn more about it. Note how the word size is “2” to account for short- sequences. It is normally 3 for protein sequences.

7. Run **BLAST**. Note that it will take a little longer this time because the database we are searching against is larger. Compare the **E-value** to that from your earlier search.

Is this E-value larger or smaller?

Q8. Compare your BLAST results between Step 4 and Step 7 above. Notice the E-values are different, but the **bit scores** remain the same. Why?

Next, we will try to infer the putative function of a protein of unknown function by identifying its *homologues* (evolutionarily related sequences) in other species in the database.

8. Retrieve the sequence **ERB27942.1** (hypothetical protein from *Escherichia coli* UMEA 3292-1) from NCBI. See Practical 1 if you do not know how.

9. In a new protein-protein BLAST (blastp) window, reset the BLAST options to default (**Reset page**; top right), and use the **ERB27942.1** sequence as query.

Tip: Instead of the actual sequence, you can simply enter the accession number or GI number.

10. Under **Algorithm parameters**, change the **Max target sequence** from 100 to **500**, and check the **Filter** for **Low complexity regions** (this will mask low complexity regions from the query before BLAST). Filtering can eliminate statistically significant but biologically uninteresting reports from the output. Set the Matrix as **BLOSUM62**. We will run BLAST against the whole **nr** database.

11. Run **BLAST**. This search may take a little longer than earlier searches because the query sequence is longer.

The top match should be the query sequence itself. Have a look at the alignment of the top match. You may see many hits from other *Escherichia coli* proteins. Do you see putative homologues from other species as well?

Note that there are lower-scoring matches (Bit score <100) form other species with E-values that suggest they are still statistically significant (e.g. those with **E-value < 1e-10**), i.e. it is unlikely one would find such matches by chance. These hits are likely homologous proteins from other species.

Looking at the *Descriptions* table, annotations for some of the higher scoring matches indicate that in addition to *E. coli*, there are known effector proteins from *Yersinia* and *Shigella*. This is a good indication that **ERB27942.1** protein too is an effector. Take particular note of **OspG** from *Shigella flexneri* (accession **EFQ1385283.1**), which is one of the few matching proteins among the hits that has actually been characterised in the laboratory. Recalling what you learned in last week's prac, this hit's GenBank page will tell you more about the publications and additional evidence associated with this protein.

Case Study 2
A paper was published about OspG protein in *Shigella flexneri* (**PMID:16162672**; recall PubMed in Prac 1). The researchers found that OspG interferes with innate immunity of the human host responses by targeting specific enzymes within the host. This effect was mediated by auto-phosphorylation activity of OspG, suggesting that OspG is a type of protein kinase. *Note: kinases are a class of protein that mediate attachment of phosphate to themselves or other molecules.* **§ Supplementary Note 8**

Is **ERB27942.1** a protein kinase too? The specific catalytic residues on OspG responsible for auto-phosphorylation were identified (Figure 2 in PMID:16162672). We can compare how these are conserved in our sequence of interest. The catalytic residues in *Shigella flexneri* OspG are found in 3 different places in the sequence (shown below; catalytic residues highlighted in underlined boldface). In the actual tertiary protein structure of OspG, these residues probably fold together to form a catalytic pocket on the surface of the protein.

I **G** Q **G** S T	L Y **K** K Y	W Q **E** S E

12. Examine the alignment of your query to the *Shigella* effector OspG sequence (**EFQ1385283.1**) on your BLAST output (*hint:* use the simple text search function on your browser to find it on the page). Do the catalytic residues match?

Q9. Write the residues in **ERB27942.1** that correspond to the 16 OspG residues above.

These catalytic residues are conserved in **ERB27942.1**. Finding conserved residues like this can be very helpful in deciding whether a BLAST match represents a biologically meaningful alignment. It is also a good example of how a BLAST result can be used to prime hypothesis-driven laboratory research. i.e. this protein may also be capable of auto-phosphorylation, and because it too is an effector, it might target a similar human target.

You have learnt the principles of the BLAST heuristic using BLASTp (protein BLAST). The other *flavours* of BLAST are simply designed for different purposes. If you want to search your protein sequence against a nucleotide/genome sequence, you should use **tBLASTn**, which searches just like BLASTp, but against a database that is made of all six-frame translations of the target nucleotide sequence.

For more information about all other BLAST algorithms, visit the BLAST help page:
http://blast.ncbi.nlm.nih.gov/Blast.cgi?PAGE_TYPE=BlastDocs

Concluding remarks

Have a look at the learning objectives at the beginning of this practical. You should now be familiar with the fundamental principles of pair-wise sequence alignment and of the dynamic programming in generating heuristic (instead of exact) alignment solutions. You should have a basic understanding of the key parameters involved in a sequence alignment, including alignment scores and gap penalties. You should also be able to use the appropriate BLAST programs to search for the sequences you are interested in from GenBank, and you should know how to interpret the results based on observed E-values and bit scores.

ANSWER KEY

Q1. 6 lines (number of tandem repeats minus one)

Q2. Inversion

Q3. A. 6; **B.** Diagonal (above-left)

Q4. A. Global alignment; **B.** Local alignment

Q5.

A.
```
AAATTTAAA
AAA---AAA
3+3+3-3-1-1+3+3+3 = 13
```

B.
```
AAATTTAAA
AAAAAA---
3+3+3-2-2-2 = 3
```
(NB. If the alignment from **A** were used: **-4**). The last three positions in **B** above do not invoke gap opening/extension.

Q6. 7. Calculation: 6 + 2 + (-1)

Q7. True. Smaller E-values are less likely to be a match simply by chance.

Q8. E-value is dependent on database size, bit score does not. In this case the two alignments are identical, so given the same search parameters they should produce identical bit score.

Q9.
```
ERB27942.1: I G R G L A
OspG:       I G Q G S T

ERB27942.1: V I K R Y
OspG:       L Y K K Y

ERB27942.1: Q K E C H
OspG:       W Q E S E
```

Practical 3: Sequence analysis and phylogenetics

In this practical, we will analyse conserved motifs in sequences, perform multiple sequence alignment and infer phylogenetic relationships. **§ Supplementary Note 1**

Learning objectives:

A. to understand patterns/motifs and pattern searches in sequences
B. to understand position-specific scoring matrices and sequence logos
C. to analyse and search for protein domains based on profile Hidden Markov Models
D. to create multiple sequence alignments and phylogenetic trees

Additional references:

- Textbook – Zvelebil NJ & Baum JO (2008) *Understanding Bioinformatics*, Garland Science:
  - *Chapter 3* Section 3.3 & 3.4 (p55-66)
  - *Chapter 4* Section 4.5: *Types of Alignment* (p87-93); Sections 4.8, 4.9 & 4.10 (p103-111)
  - *Chapter 6* Patterns, Profiles, and Multiple Alignments (p165-220)
  - *Chapter 7* Recovering Evolutionary History (p223-266)
  - *Chapter 8* Building Phylogenetic Trees (p267-313)
- See also **Practical 3 Supplementary Notes** and **Videos** on Blackboard

A. Patterns/motifs in sequences

A pattern or motif in bioinformatics is a way of describing functional regions of proteins or genes; for example, regions encoding binding sites or protein domains. These regions may be defined experimentally in the lab and/or by comparison of many related sequences from different species. Once defined, patterns or motifs can be used to identify these functional regions in unknown or novel sequences.

Here we will learn the common notation for a protein pattern. Given a hypothetical protein sequence fragment **WQEEESL**, this sequence could be described by the following notation:

```
[TW]-x-E(3)-[VS]-x(2)
```

This notation is similar to that used in regular expression notation in e.g. UNIX.

```
1st amino acid: T or W
2nd amino acid: any
3rd-5th amino acid: EEE (exactly 3 Es), simplified from E-E-E
6th amino acid: V or S
7th (and 8th) amino acids: any two
Dashes are added in between positions
```

Now consider this pattern: `[TW]-x(3)-E-{MFLC}-x(3,5)`

```
1st amino acid: T or W
2nd-4th amino acid: any three
5th amino acid: E
6th amino acid: any amino acid BUT M, F, L or C
7th (and following): at least 3, at most 5 of any amino acid
```

Although both patterns represent the same segment of sequence, the first pattern is **more specific** than second. Searching these patterns against the databases would generate different results.

For more information on the notation, refer to your textbook (Table 4.2, pg 108), or for instance:
http://pir.georgetown.edu/pirwww/support/help.shtml#8.1

Q1. Consider the region in *Shigella flexneri* OspG protein that encompasses the first two catalytic sites: **LIGQGSTAEIFEDVNDSSALYKKY**. Below is a multiple sequence alignment of 5 sequences at this region (residues conserved across all sequences are highlighted in red).

```
seq1 VIGKGGNAVVYEDMDDTTKVLKMF
seq2 VIGKGGNAVVYEDAEDATKVLKMF
seq3 VIGKGGNAVVYEDAEDATKVLKMF
seq4 VIGKGGNAVVYEDMEDTTKVLKMF
seq5 LIGQGSTAEIFEDVNDSSALYKKY
```

A. Construct a pattern for this region based on this alignment.
B. Construct a pattern for this region, now replacing/simplifying all ambiguous positions with the wildcard character **x**, in the simplest form. *The first three positions are done for you below.*

A. `[VL]-I-G-`

B. `x-I-G-`

3. Try searching your patterns from **Q1** against the comprehensive **UniProtKB** database using ScanProsite (http://prosite.expasy.org/scanprosite/). Use **Option 2** (*Submit MOTIFs …*) default parameters. It may take a few minutes to receive the results.

What similarities and differences do you observe between the results found with the two patterns? Do you observe the same protein/s coming up in the two results? If so, how do the E-values differ? Why do you think this might be the case?

For your convenience, the results from the PIR searches using the two patterns are also available on Blackboard (**`Prac3_patternA-output.pdf`** and **`Prac3_patternB-output.pdf`**).

B1. Position-specific scoring matrices (PSSMs)

Unknown sequences can be characterised by identifying features, e.g. protein domains, within their sequences. To do this, such unknown sequences can be searched against databases of known patterns and profiles, e.g. using **InterProScan** (https://www.ebi.ac.uk/interpro/search/sequence/). But how are these patterns and profiles derived in the first place? Here we will examine the sequence information that defines the conserved zinc finger domain (NCBI Conserved Domain **cd02249**).

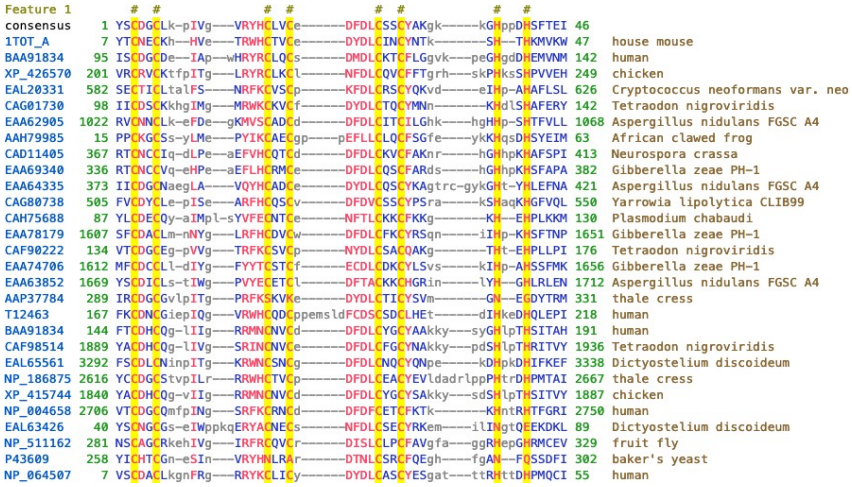


FIGURE 1

1. View entry **cd02249** on NCBI's **Conserved Domain Database** (https://www.ncbi.nlm.nih.gov/cdd/).

2. View the multiple sequence alignment associated with this domain at the bottom of the page.

You should see something like in **Figure 1.** Show all the rows with the **Row Display** toggle and check the **Show Consensus Sequence** box.

This alignment shows the zinc finger domain regions (positions in green) from an assortment of proteins (accession numbers in blue) from different species (listed at the far right). Residues that are identical in all sequences are highlighted in yellow and marked with a #. The consensus row does not represent a real sequence – instead, this summarises the most common amino acid found at each position.

This alignment can also be represented as a position-specific scoring matrix (PSSM). PSSMs are constructed from highly-conserved regions of multiple sequence alignments. **Profiles** are PSSMs that allow for position-specific gaps. PSSMs are similar to substitution matrices (Practical 2 Figure 8), but instead of specifying a single score for each amino acid substitution, a PSSM specifies different scores for each amino acid for every specific position in the alignment. Thus, an unknown sequence can be scored based on biologically-relevant observations of a given region.

FIGURE 2

A P - consensus sequence position B C - consensus sequence residue

Master: 1TOT_A C View Master in CD

P	C	Master	A	G	I	L	V	M	F	W	P	C	S	T	Y	N	Q	H	K	R	D	E
1	Y	7 - Y	-5	-6	3	-4	2	1	4	-3	0	-5	0	-4	6	0	-5	-4	-5	2	-6	-5
2	S	8 - T	-1	-5	2	-2	0	-4	1	-6	0	1	3	3	-4	-4	-4	-5	-1	1	-5	-1
3	C	9 - C	-4	-6	-5	-5	-4	-5	-6	-6	-6	12	-4	-4	-6	-6	-7	-7	-7	-7	-7	-7
4	D	10 - N	-1	-5	-7	-7	-6	-6	-7	-7	-5	-7	-3	-1	-6	4	-3	3	0	-1	7	-2
5	G	11 - E	0	4	1	-2	-2	-5	-5	-6	-6	5	-2	0	0	2	-4	3	-5	-5	-5	0
6	C	12 - C	-4	-6	-5	-5	-4	-5	-6	-6	-6	12	-4	-4	-6	-6	-7	-7	-7	-7	-7	-7
7	L	13 - K	-4	3	-1	3	-1	-3	-5	-6	-5	-6	0	-4	-5	1	3	-5	1	0	-1	-1
8	K	14 - H	-2	1	1	-2	-1	3	-5	-6	-5	-5	2	2	0	1	1	1	2	-4	-4	0
9	P	Gap	-2	0	-4	0	-2	-4	-5	-5	5	-5	-3	-1	1	1	-3	2	-4	-4	1	3
10	I	15 - H	-5	-7	7	1	0	-2	3	-5	-6	-5	0	-4	-4	-1	-6	3	-6	-6	-6	-6

Figure 2 shows the first 10 positions of the PSSM for the conserved zinc finger domain. The first two columns, P and C, represent the consensus sequence position **(A)** and residue **(B)** within the pattern. These are identical to the consensus sequence shown in **Figure 1.** The third column, Master, shows one of the true sequences from the alignment **(C)** – this could be any sequence; in this case, it is 1TOT_A, a house mouse sequence also found in the first row of the alignment in Figure 1.

The remaining 20 columns (A – E) represent each of the 20 amino acids. These scores are similar to the log-odds scores (Practical 2). The difference here is that each amino acid (column) score is assigned (weighted) depending on its position (row). The greater the score for an amino acid at a specific position, the more likely one would find that amino acid in that position. The total score of any sequence can be found by taking the sum of the scores across their corresponding positions.

Consider the first 7 consensus sequence residues of the domain in the PSSM. Notice the all-cysteine positions (3 and 6). You should see that **C** has the highest score in these positions (Figure 2) – in this case, they both have a score of 12.

You can see why BLAST is not a very useful tool in recovering these matching patterns/motifs across a set of sequences. These motifs are generally short, with few conserved residues interspersed between stretches of non-conserved residues.

Q2. Based on the PSSM and alignment in **Figures 1** and **2**, find the following information.

(a) The total PSSM score for positions 7-13 in the **`Master`** (i.e. corresponding to the first 7 *consensus sequence residues*)

(b) Using the maximal PSSM scores for each position, construct the least ambiguous pattern describing the first 7 *consensus sequence residues*.

(c) Based on the pattern you constructed in **(b)**, can you find any sequence(s) in the alignment in **Figure 2** that has the first 7 amino acids matching your pattern?

(d) Consider the first three sequences in the alignment (**`1TOT_A`**, **`BAA91834`**, and **`XP_426570`**). Based on the PSSM scores across the first 7 *consensus sequence positions*, which of the three has the lowest score?

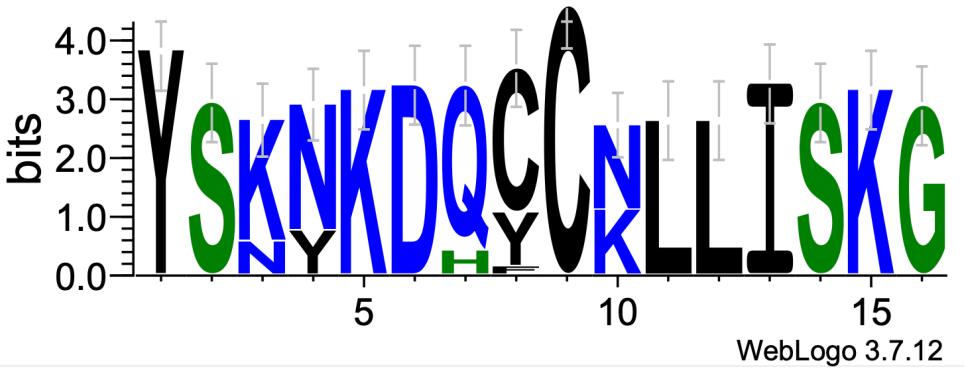
B2. Sequence logos § Supplementary Note 2

Another way to graphically summarise information in a multiple sequence alignment is with a sequence logo. Here, the observed residues in each position are stacked, with the height of each letter proportional to its frequency or score. Sequence logos are a simple visual way to quickly assess an alignment. Here, you will learn how to generate simple sequence logos based on either consensus or PSSM. The file **`Prac3_sopE_signature.fasta.txt`** (on Blackboard) contains a 16-amino acid signature within the *Salmonella* SopE protein, from 23 sequences.

1. Using **WebLogo 3** (http://weblogo.threeplusone.com/), create a **consensus sequence logo** for this signature, using the 23 sequences as input. Create the logo using default settings.

Your sequence logo should look the same as **Figure 3**. The overall height of the stack indicates the sequence conservation at that position (i.e. how conserved is the position), while the height of symbols within the stack indicates the relative frequency of each amino acid at that position. By default, residues are coloured according to amino acid properties (hydrophilic = blue, hydrophobic = black, neutral = green).

FIGURE 3



More info: http://weblogo.threeplusone.com/manual.html

Q3. Based on your consensus sequence logo, find the following information.

(a) The number of unambiguous positions in this signature.

(b) The most conserved position, and its corresponding amino acid.

(c) The most ambiguous position, and the different amino acids found at this position.

(d) The number of positions that contain two possible amino acids.

(e) The overall consensus signature based on the most likely amino acid in each position.

2. Using **Seq2Logo** (https://services.healthtech.dtu.dk/service.php?Seq2Logo-2.0), create a PSSM-logo for this signature using the 23 sequences as input (default settings except **Logo type** set as **PSSM-Logo**).

This is a logo based on the PSSM as calculated from the sequences. Unlike the previous sequence logo, here all 20 amino acids are shown at each position, many with negative bits values, suggesting that the residues are unlikely to occur at the corresponding positions. For instance, at position 9 only **C** has a positive value (an amino acid other than **C** is very unlikely to occur at position 9).

3. Note down the most likely amino acid at each position. This signature should be the same as your consensus signature in **Q3(e)** above.

Q4. Based on your PSSM-logo, find the following information.

(a) The least likely amino acid at position 9 of this signature.

(b) The hydrophobic positions after position 9 in this signature (*Hint: Leucine and isoleucine are examples of hydrophobic amino acids*).

(c) The positions that contain only one amino acid with a positive bits value.

As you can see, there are many other algorithms available to display a sequence logo. If you have some time, read about the different SeqLogo **Output formats** (there is link on the pink panel above *Submission* at the Seq2Logo server site, or test them out with your data).

C. Domain structures and profile Hidden Markov Models (HMM)

InterPro is a database of protein features (signatures) such as domains and functional sites. It is very comprehensive because it combines information from multiple member databases (including the Conserved Domain Database and Pfam, both of which we also use in this Practical). Here, we use InterPro to examine domain organisation of a protein. More info: https://interpro-documentation.readthedocs.io/en/latest/ **§ Supplementary Note 3**

1. Search for **SopE** on **InterPro** within the **Browse** tab (`http://www.ebi.ac.uk/interpro/`).

Q5. Navigate through the InterPro site to find the following information about SopE. Hint: the filters within the Browse tab will be helpful.

(a) How many SopE families did you find with your search? Which InterPro identifier represents the SopE family?

(b) What is the protein that reverses the actions brought about by SopE?

(c) How many different types of protein domain are observed in the SopE family?

(d) Based on **(c)**, how many proteins are found to contain both N-terminal and GEF domains?

(e) Of the proteins in **(d)**, what is the InterPro accession number for the first protein in **`Salmonella typhimurium`** on the list?

(f) How many three-dimensional structures of SopE family are found in the database?

(g) Based on annotation of Gene Ontology (GO) terms of the SopE family, where is this protein located in the cell, i.e. what is/are the term(s) associated with *Cellular Component*?

Next we will investigate the SopE protein family further. We will examine one of the domains within the protein you identified in *Salmonella typhimurium* (in Q5(e) above). A legacy database of protein domains is Pfam, which is now fully integrated within InterPro. **§ Supplementary Note 4**

2. On InterPro, retrieve the protein record for the accession number you identified in **Q5(e)**.

You should see that the protein contains two domains: the N-terminal domain (IPR016018; Pfam identifier PF05364), and and the GEF domain (IPR016019; Pfam identifier PF07487). Let's examine the GEF domain in greater detail.

Q6. Navigate through the InterPro record of **IPR016019** to find the following information about this domain.

(a) In how many unique domain architectures is the GEF found?

(b) How many Alphafold-predicted tertiary structures are available for this domain?

(c) Is this domain found in viruses?

(d) What is the most represented genus among the proteomes in which the domain is found?

(e) How many GEF-containing sequences are described in *Gammaproteobacteria*?

(f) Among bacterial species in which GEF-containing sequences are found, how many are classified as *Enterobacteriaceae*?

(g) What is the PDB identifier for the resolved protein structure showing the complex between the GEF domain of the *Salmonella typhimurium* SopE protein and the human protein of Cdc42?

D1. Multiple sequence alignment

Above, we observed existing multiple sequence alignments used to define known protein domains. Here we will use **MEGA** to generate a multiple sequence alignment, and perform additional phylogenetic analysis using a set of OspG sequences from different species (**Prac3_OspG_seqs.fasta** on Blackboard). **§ Supplementary Note 5**

MEGA is installed on all iLC1 computers. The software is also freely available at http://www.megasoftware.net/.

1. Open the sequence file in **MEGA**. In the MEGA main window, **Data → Open A File/Session**, select the file where it is located. We want to **align** the sequences, not **analyse**.

2. You should see 5 sequences related to the OspG proteins in **Alignment Explorer** (new pop-up window). These sequences have not been aligned yet. Two multiple sequence alignment (MSA) programs are available: **ClustalW** and **MUSCLE**; note the **W and arm icons** and the **Alignment** menu.

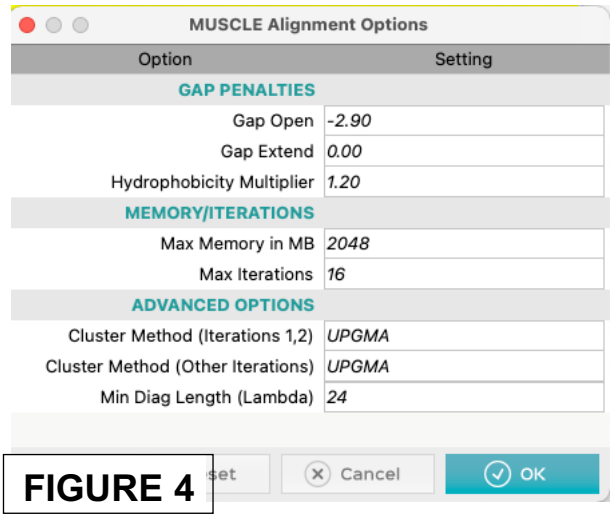


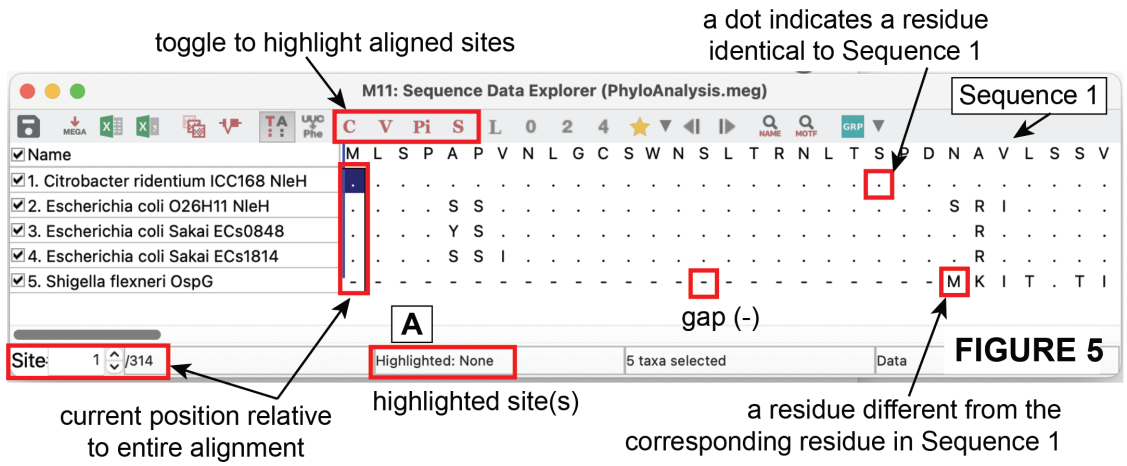
FIGURE 4

3. We will use **MUSCLE** to align these protein sequences. Select all protein sequences and run MUSCLE. Note the available parameters (**Figure 4**) and recall gap penalties from Practical 2. Run MUSCLE using the default setting. Proceed with **OK**.

Have a look at your MSA. *Does the placement of gaps make any sense to you?* Notice the position number at the bottom left of the window as you click on an aligned column/position. Note the columns with an asterisk (*) above. *What are these columns? Get a rough estimate of the number of these columns.*

4. Let's view the alignment in a simpler form. In the **top menu bar,** go **Data → Phylogenetic Analysis**. If prompted *Protein-coding nucleotide sequence data?* click **Yes**.

You will see a new icon  on your main (original) MEGA window. Click on the icon to pop up the **Sequence Data Explorer** (as shown in **Figure 5**).



The first sequence is shown on top as a consensus sequence. Note the toggles to highlight specific aligned sites (positions) in the alignment: conserved (**C**), variable (**V**), parsimoniously informative (**Pi**) and singleton (**S**) sites. When clicked, the corresponding sites are highlighted in yellow, and their proportion is shown (**Figure 5A**).

Click through each toggle and try to understand what each of them represents. For instance, a site is *parsimoniously informative* if it contains ≥2 distinct residues, and ≥2 of them occur ≥2 times, i.e. this site contains information sufficient for us to convincingly tell them apart into two or more groups. *You can find out more using **Help**.*

Q7. Based on your MUSCLE alignment ran at default settings, find the following information.

(a) What is the total length of the MSA (how many aligned positions are there)?

(b) How many of these aligned positions are conserved?

(c) How many of these aligned positions are parsimoniously informative?

(d) Of all parsimoniously informative sites in the MSA, how many of those positions are missing from the *Shigella flexneri* sequence?

(e) Does this MSA contain more singleton sites than the variable sites?

(f) In which sequence can you find this motif: **`LIGQGSTAEIFEDVNDSSALYKKY`**? How many conserved positions are found within this motif in the alignment?

Leave all your MEGA windows open for the next exercise.

D2. Phylogenetic inference § Supplementary Note 6

A **phylogenetic tree** is a diagram that depicts the evolutionary relationships among different species/entities based upon the similarities and differences in their genetic characteristics.

Here we will infer a maximum likelihood (**ML**) tree using the MSA above (generated using MUSCLE at default settings). An ML approach searches for *the* tree that is most likely describing the relationships among the sequences, given an explicit evolutionary model and the data.

1. On the main MEGA window, go **Phylogeny** > construct/test ML tree. Use *currently active data*.

2. Set the parameter settings for ML analysis in the **Analysis Preferences** window. Use the settings shown in **Figure 6**. Proceed with **OK**.

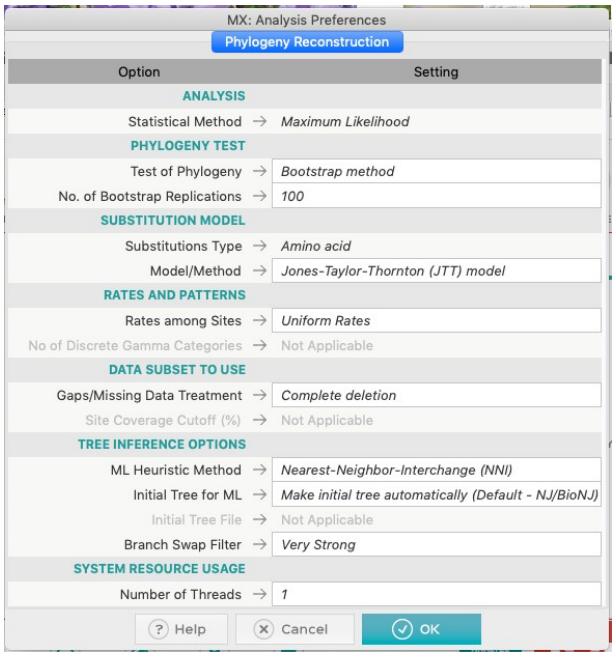


FIGURE 6

The specification of a substitution model is a **MUST** in ML approaches. The JTT model is a commonly used model that looks similar to the substitution matrix you had in Practical 2 when you did the RAT-and-RAM comparison.

3. Your optimal ML tree should pop up in the **Tree Explorer** window, in the **Bootstrap tree** view. Your tree might look a little different to the tree in **Figure 7**.

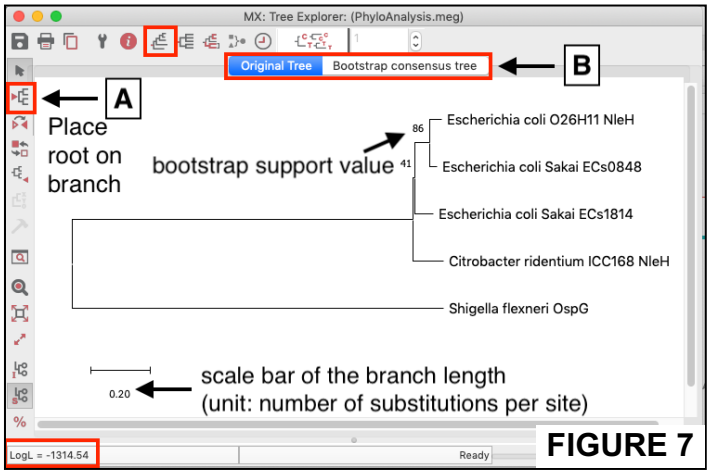


FIGURE 7

4. On your Original Tree, you may view/hide the branch lengths: **View** → **Show/Hide** → **Branch Lengths**. *Why can you not view/hide the branch lengths on the Bootstrap consensus tree?*

In the tree in **Figure 7**, the clade of *Escherichia coli* O26:H11 NleH and *E. coli* Sakai ECs0048 is supported by 86% of the sampled trees (in this case, 86 of 100 samples) in the bootstrap analysis, suggesting a close relationship between the two, given the data and substitution model. This also means that in the remaining 14% of the samples, one of these two branches was found to group with other branches (*i.e.* the branch is located elsewhere in the tree).

5. To root the tree using a tree branch, click on the **Place root on branch tool** (**Figure 7A**), and the branch.

This will root the tree using that branch (ideally this is the **outgroup**). **§ Supplementary Note 6**

You can also swap/collapse the branches using the other tools on the toolbar. Spend some time to see what each of these tools does. Explore the information icon and the different tree structures (**Figure 7B**).

6. Leave this tree open, and re-construct another ML tree using the same parameter settings.

Does your tree look the same as before? Why not?

Q8. Based on your tree from step 5, find the following information.

(a) Based on your Original tree, which of the five sequences is the appropriate outgroup?

(b) Rooting the Original tree using sequence in **(a)** as outgroup, what are the lengths of the longest and shortest branches?

(c) Based on your Bootstrap consensus tree, what is the bootstrap support for the *Escherichia coli* clade?

(d) Which two of the five sequences are paralogues?

(e) What is the log likelihood of the Original tree?

(f) Construct another ML tree using the same MSA and parameter setting, but now using the **WAG** substitution (evolutionary) model instead of JTT. What is the log likelihood of this tree?

(g) Based on your answers in **(e)** and **(f)**, which evolutionary model is a better fit for this MSA?

Concluding remarks

You should now be familiar with motifs and pattern searches, PSSM and sequence logos. You should be able to navigate and access necessary information from the different databases of protein (and domain) families. You should have a basic understanding of MSA and phylogenetics, and be able to construct MSA, infer and interpret phylogenetic relationships among a set of homologous sequences.

This concludes the core practicals. **Refer to Page 1 of this document for details and due date of Assessment 1 associated with these practicals.**

ANSWER KEY

Q1 A. `[VL]-I-G-[KQ]-G-[GS]-[NT]-A-[EV]-[IV]-[FY]-E-D-[AMV]-[DEN]-D-[AST]-[ST]-[AK]-[LV]-[LY]-K-[KM]-[FY]`
B. `x-I-G-x-G-x(2)-A-x(3)-E-D-x(2)-D-x(5)-K-x(2)`

Q2. **(a)** 38 (6 + 3 + 12 + 4 + 0 + 12 + 1)
(b) `Y-[ST]-C-D-C-C-[GLQ]`
(c) No.
(d) gi50750334. *This sequence has the lowest score.*
Master/1TOT_A (`YTCNECK`): 38 (see Q2(a) above)
BAA91834 (`ISCDGCD`): 3 + 3 + 12 + 7 + 4 + 12 + (-1) = 40
XP_426570 (`VRCRVCK`): 2 + 1 + 12 + (-1) + (-2) + 12 + 1 = 25

Q3. **(a)** 11 **(b)** position 9, C (cysteine) **(c)** position 8, C (cysteine), Y (tyrosine) and F (phenylalanine)
(d) 4 **(e)** `YSKNKDQCCNLLISKG`

Q4. **(a)** E (glutamic acid) **(b)** positions 11-13 **(c)** positions 9 and 16

Q5. **(a)** 1, IPR005414 **(b)** tyrosine phosphatase effector (SptP) **(c)** 2 (IPR016018 and IPR016019)
(d) 853* **(e)** A0A0D6FRJ4* **(f)** 5 **(g)** extracellular region
*: these answers may change as the database expands

Q6. **(a)** 2 **(b)** 971 **(c)** No **(d)** *Salmonella*
(e) 967 **(f)** 269 **(g)** 1gzs

Q7. **(a)** 314 positions **(b)** 122 **(c)** 18 **(d)** 5
(e) No, there are more Variable (171) than Singleton sites (153)
(f) *Shigella flexneri* OspG sequence, 8 conserved positions

Q8. **(a)** *Shigella flexneri* OspG sequence **(b)** 0.98, 0.02 substitutions/site **(c)** 59%
(d) *E. coli* Sakai ECs0848 and *E. coli* Sakai ECs1814 (the two sequences from the Sakai genome)
(e) -1311.10 **(f)** -1315.97 **(g)** JTT model is a better fit than WAG model for this MSA.

There is no definite answer for Q8 (b), (c), (e) and (f). The key is for you to understand the concepts of branch length, bootstrap and log likelihood, and to interpret a phylogenetic tree. Each ML tree will be slightly different from each other, but the main conclusions drawn from the tree should be the same.