Introduction to Bioinformatics
6th practice
DATABASE SEARCH - BLAST

During this session you will practice how to use a database search program such as Blast. You will partly work with the well known LasR. You will practice protein/gene identification in different database types, investigate what the key factors affecting significance are and how to evaluate rankings.

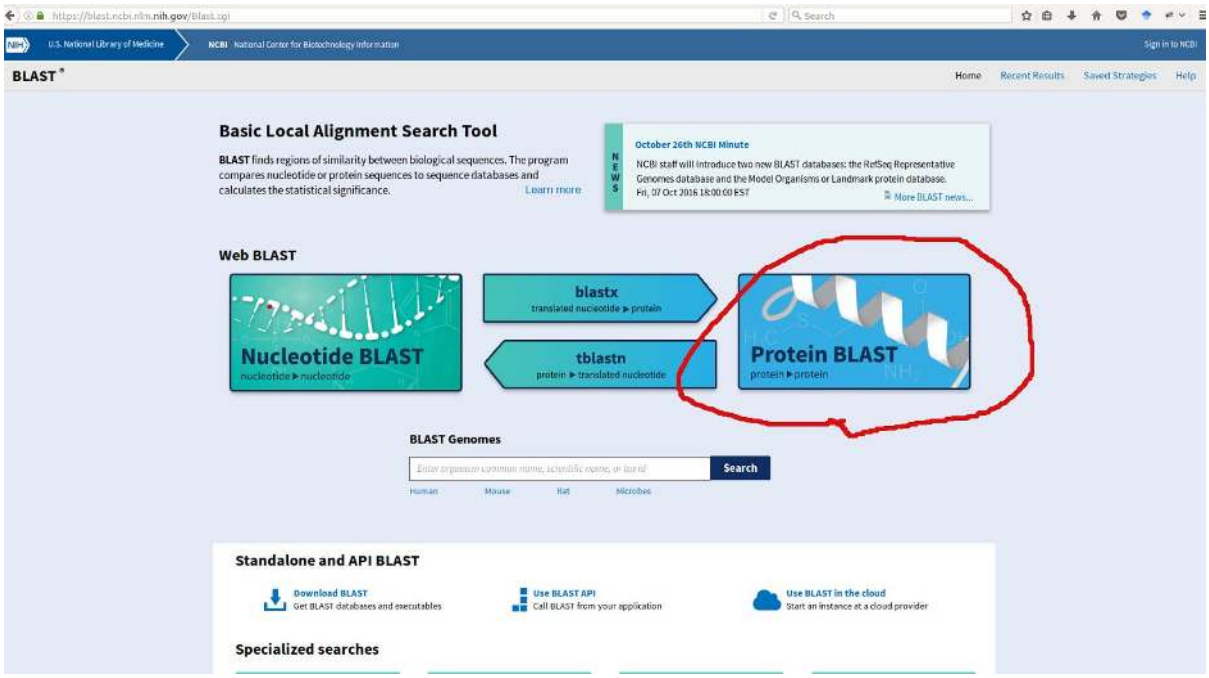
The required software is only a browser using the site:
https://blast.ncbi.nlm.nih.gov/Blast.cgi (or http://www.uniprot.org/blast/).

Files you need:

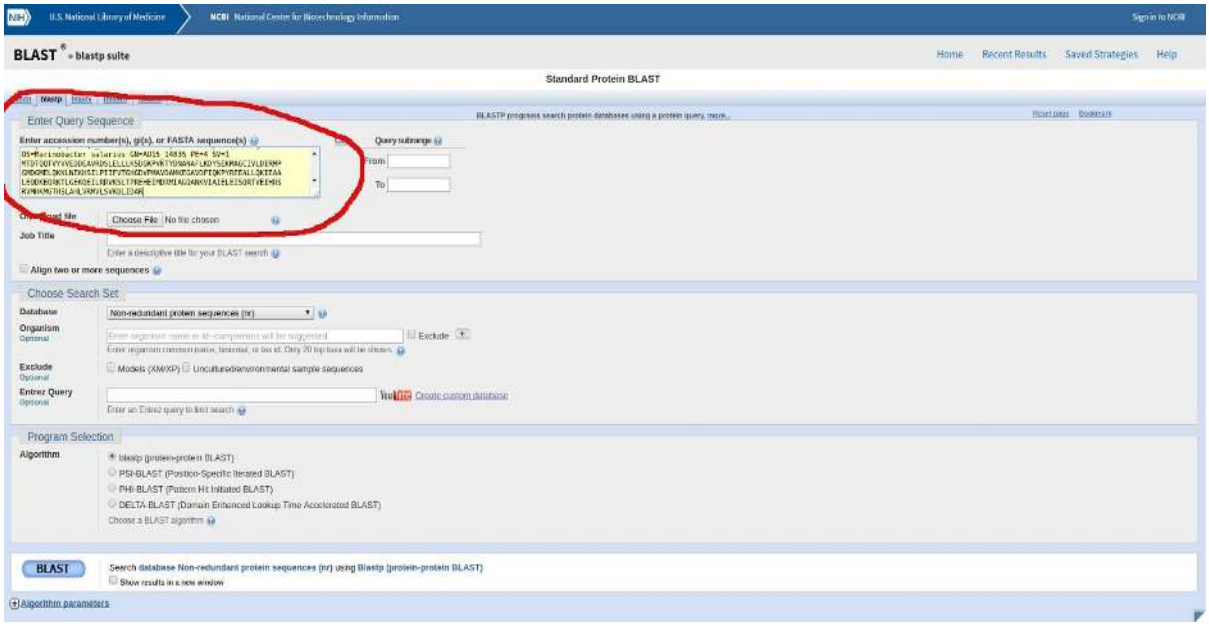
1. **blast_protein_sequences.fasta**
2. **blast_nucleotide_sequences.fasta**

Database searching:
BLAST is one of the most famous sequence searching programs. It gives you information on non-random similarities between biological sequences. The non-chance similarities arise due to homology.

Selecting program
The search starts with a sequence. This could be either a protein or a nucleotide. In this part we are going to work with a protein sequence.
Visit the BLAST site at the NCBI and choose the protein blast:
https://blast.ncbi.nlm.nih.gov/Blast.cgi



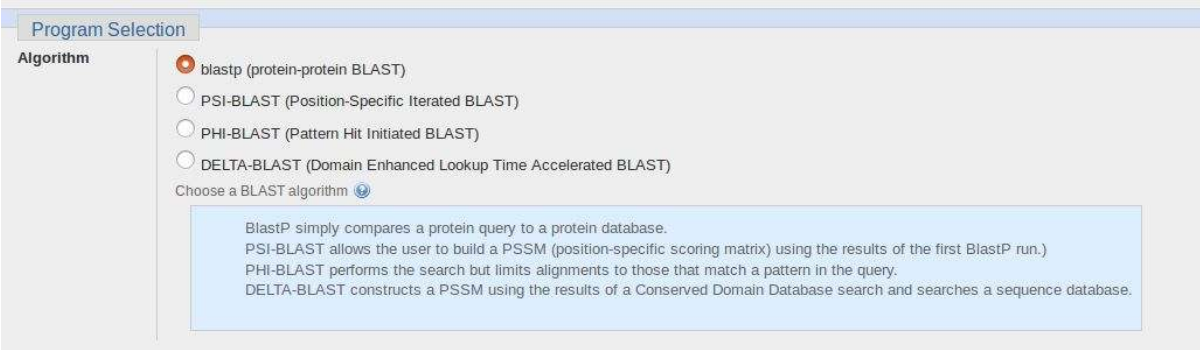
Then download **blast_protein_sequences.fasta from wiki page** and copy the sequence of LasR protein with fasta header: `>tr|W5Z0Q8|W5Z0Q8_9ALTE LuxR family transcriptional regulator OS=Marinobacter salarius GN=AU15_14835 PE=4 SV=1` into the “Enter Query Sequence”.



Selecting database
The next step is choosing a sequence database to compare to.
The default database is nr (~non-redundant), which is a comprehensive database covering GenBank CDS translations, PDB, SwissProt, PIR and PRF. However, nr does not contain proteins from metagenomes, next-generation assemblies and patents. Other useful subsets are Swiss-Prot, PDB.

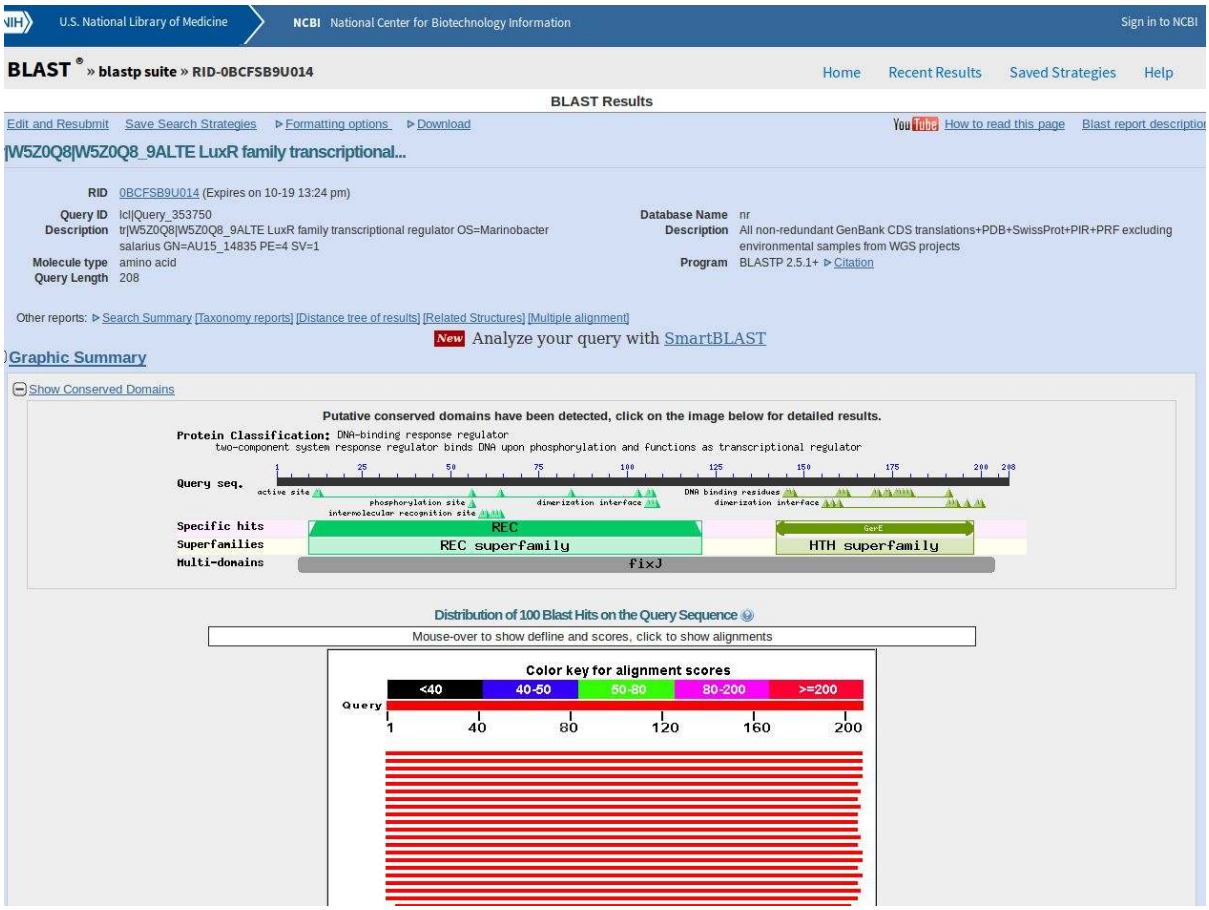
There are other options to define your own search space via the proper database ids or entrez search. It is useful if you may want to search only in bacteria sequences or in staphylococcus genomes.

Choosing algorithm
There are various different algorithms for database search tailored to specific tasks:

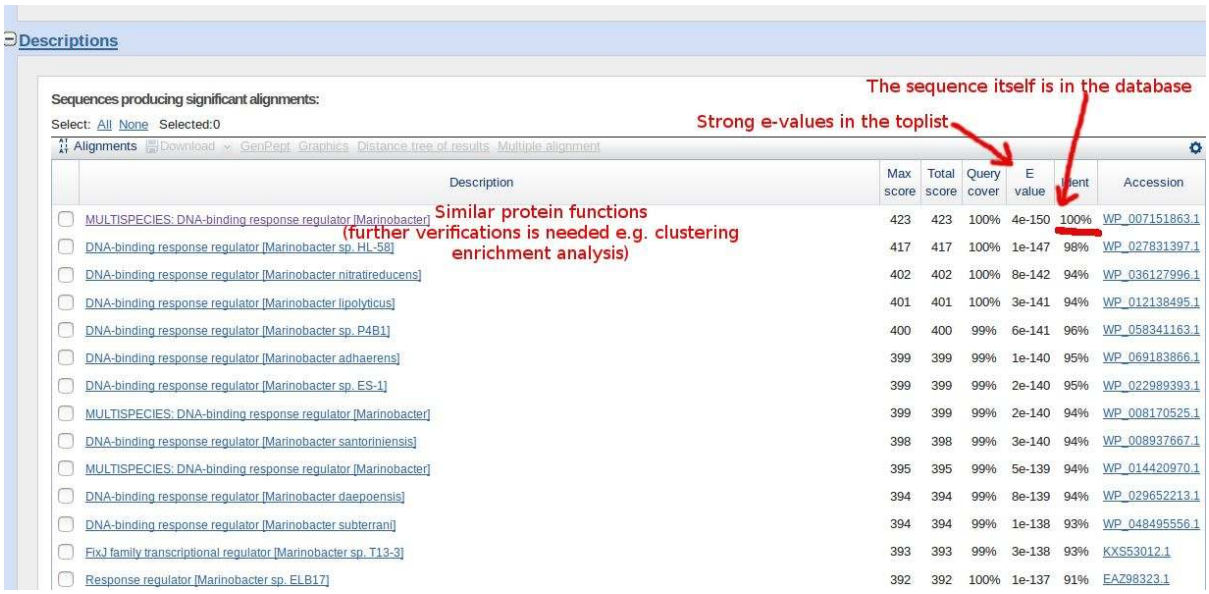


blastp algorithm is the original, that works as presented in the lecture.

If you set everything, hit the BLAST button. Be aware that the database search can take a while.
You will get the following results.



In the graphics summary you may immediately see the two domains in the query proteins.



Exercises

Repeat the BLAST search with the sequence `>suffled_tr|W5Z0Q8|W5Z0Q8_9ALTE` in which the two domains are switched. Filter the database into *marinobacter* sequences (the taxon id of *marinobacter* is 2742). What do you expect as a result?

Repeat the protein blast with the domains separately (sequences with header "`>tr|W5Z0Q8|137-202 GerE`" and "`>tr|W5Z0Q8|7-121 Autoinducer_domain_REC`"). Look the "scales" of the e-values. What do you experience, how the e-values vary depending on the length of the query? What do you think the number counting identities in the alignment is a good index for measuring the quality of alignments?

Another important parameter is the seed size. If it is small then the algorithm is more sensitive but slower, since the search space is significantly larger. On the other hand, if the seed size is large, than the search space is greatly reduced, thus the running time is smaller, however the expected number of false negative hits is larger. It is valid and justifiable approach if one is looking for the strong similarities.

Find the odd one out! Which one is a random sequence and why: `>Sequence1`, `>Sequence3` or `>Sequence4`? What do you think how a typical "random match" looks like?

Nucleotide BLAST

The blastn program is the traditional BLAST algorithm, which is the most sensitive nucleotide search. If you need high sensitivity, then go with it.

On the other hand, megablast (default) uses larger word size than blastn, thus it is much faster. It also uses a different gapping model. The 2 versions are the:

1. Contiguous megablast - that is good for finding nearly identical sequences
2. Discontiguous megablast - which is useful in cross-species comparisons

Translating searches is also useful for exploring unannotated protein coding regions. The program does all the six frame translations of query, database or both. The programs are:

1. blastx – translated query
2. tblastn – translated database
3. tblastx – translated query and database

Default database (nt) is not comprehensive, it only covers a small subset of all known sequences. It does not contain bulk sequences (EST, GSS, HTGS, STS), RefSeq Genomic Sequences (Chromosome, RefSeq Genomic), Transcriptome Shotgun Assemblies.

Exercises

You have sequenced a new *marinobacter* genome and an annotation tool tagged the following sequence as hypothetical protein, but based on some evidence you found the sequence interesting. Do BLAST search with the sequence.
What kind of blast program would you choose, if you are interested in what the function of your protein could be (you may carry out multiple searches ...)? How would you optimize your program (megablast, dc-megablast, blastn)?

Analyze rankings:

The ROC analysis is widely used to evaluate the performance of a classification process. In our example, the classifier is the blast, which can be seen as a simple nearest neighbour classifier. However, the output is ranking, usually defined on the e-values (that describes the significance).

Given two ranking, where the positive elements are signed green:. Plot the ROC curve (False-positive rates (FPR) against the true-positive-rates (TPR))!
Which one is the better ranker? (The input is the same, we can assume that the algorithm was parameterized in a different way)

Algorithm_1	**Algorithm_2**
1.00E-35	1.00E-52
1.00E-34	1.00E-39
1.00E-33	1.00E-36
1.00E-29	1.00E-28
1.00E-29	1.00E-27
1.00E-22	1.00E-26
1.00E-14	1.00E-22
1.00E-12	1.00E-19
1.00E-06	1.00E-14
1.00E-05	1.00E-05