Homework #9

For the GH5 proteins in Homework #8, convert the downloaded genpept file to fasta format file using command line seqret (EMBOSS package).

Select five proteins in the fasta format file and save as a file (vi or nano) and use this smaller file as query to BLASTP search against cow rumen metagenome peptide database (/home/yyin/work/class/metagenemark_predictions.faa); save as tabular format output (use e-value < 1e-2 as cutoff, also use the -b option to change the default value to a higher number).

Use ssearch36 (/home/yyin/work/class/fasta-36.3.5e/bin/) to do the search as well (use these options: -m 8C -d 0 -E 1e-2; read the fasta manual page 9 to 11 to find out what these options mean); save as tabular format output.

Download the GH5 HMM (http://cys.bios.niu.edu/dbCAN/family.php?ID=GH5) from dbCAN and use hmmsearch to search against the cow rumen metagenomes (also use e-value < 1e-2 as cutoff); save as tabular format output.

Design command line to use grep, awk, cut, sort, uniq to process the above three output files and save hit IDs as files; wc each files to see which search method give the most hits

Office hour:

Tue, Thu and Fri 2-4pm, MO325A

Or email: yyin@niu.edu

Report due April 16 (send by email)

do the following in /media/DATAPART1/z1576493/class/mar19/

```
formatdb -i ecoli-all.faa
formatdb - # see the options, for nt db, also use -p F
less ecoli-all.faa # select the 3rd protein sequence(YP_488309.1)
vi test-query.fa # create a file to store this protein seq

[now blast, which is in your path already]
blastall -p blastp -i test-query.fa -d ecoli-all.faa
blastall -p blastp -i test-query.fa -d ecoli-all.faa > test-query.fa.out

[-m 9, the tabular format output without alignment, easy to parse]
blastall -p blastp -i test-query.fa -d ecoli-all.faa -m 9
blastall -p blastp -i test-query.fa -d ecoli-all.faa -m 9 > test-
query.fa.out.m9

[-e 1e-2, showing only hits with evalue < 1e-2]
blastall -p blastp -i test-query.fa -d ecoli-all.faa -m 9 -e 1e-2

[Now try something big (and slow)]
time blastall -p blastp -i test-query.fa -d
/home/yyin/work/class/metagenemark_predictions.faa -m 9 -e 1e-2 > test-
qery.fa.cowrumen.out.m9 &

[Do some parsing]
less test-query.fa.cowrument.out.m9 | cut -f1,2,3,7- | less
less test-query.fa.cowrument.out.m9 | cut -f1,2,3,7- | grep -v '^#' |
cut -f2 | sort -u | head
```

If a program (e.g. BLAST) runs so long on a remote Linux machine that it won't finish before you leave for home …

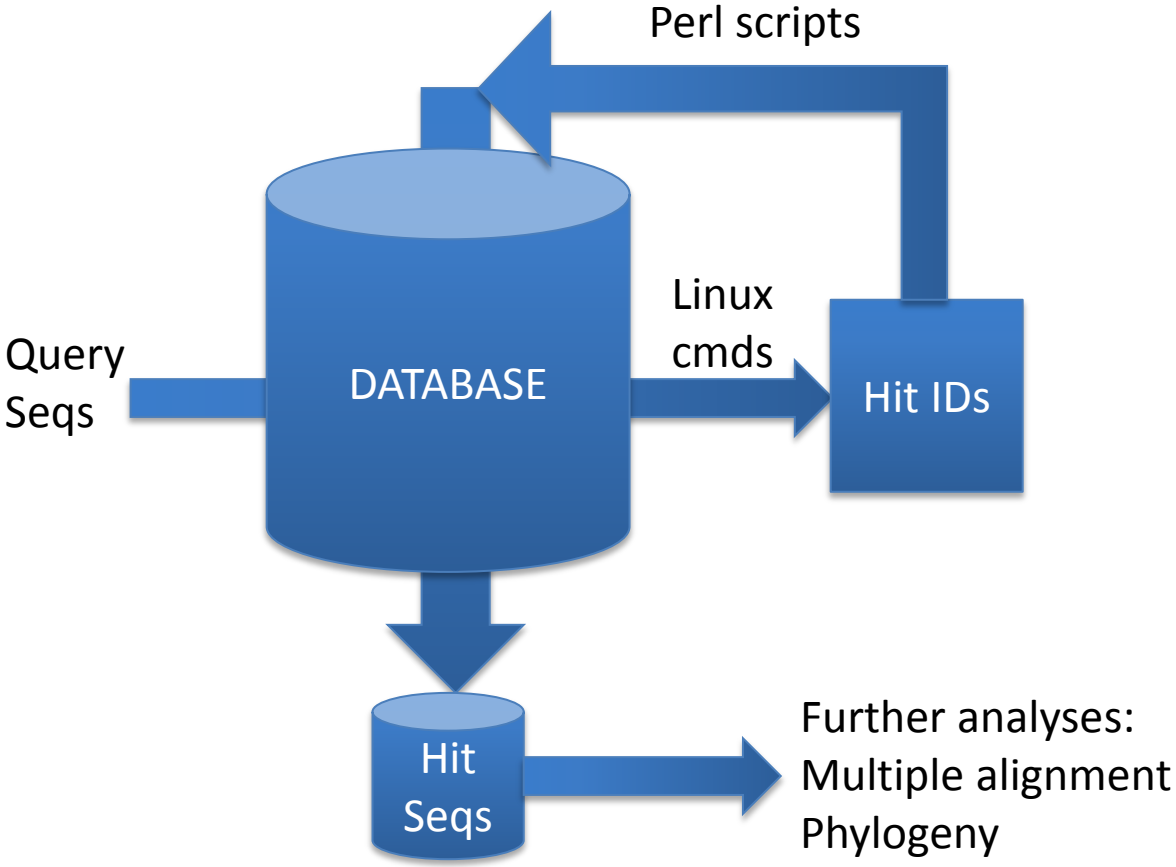
Or if you somehow want to restart your laptop/desktop where you have a Putty session is running (Windows) or a shell terminal is running (Ubuntu) …

In any case, you have to close the terminal session (or have it be automatically terminated by the server). If this happens, your program will be terminated without finishing. If you expect your program will run for a very long time, e.g. longer than 10 hours, you may put "nohup" before your command; this ensures that even if you close the terminal, the program will still run in the background until it is finished and you can log in again the next day to check the output. For example:

```
nohup blastall -p blastp -i test-query.fa –d
/home/yyin/work/class/metagenemark_predictions.faa -m 9 -e 1e-2
> test-qery.fa.cowrumen.out.m9 &
```

You will get an additional file nohup.out in the working folder and this file will be empty if nothing wrong happened.

How do you extract the sequences of the blast hits?



Multiple sequence alignment: run mafft using command line

```
mafft -h
```

```
/usr/local/bin/mafft: Cannot open --help.
-------------------------------------------------------------------------
  MAFFT v6.955b (2012/11/20)
  http://mafft.cbrc.jp/alignment/software/
  NAR 30:3059-3066 (2002), Briefings in Bioinformatics 9:286-298 (2008)
-------------------------------------------------------------------------
High speed:
  % mafft in > out
  % mafft --retree 1 in > out (fast)

High accuracy (for <~200 sequences x <~2,000 aa/nt):
  % mafft --maxiterate 1000 --localpair  in > out (% linsi in > out is also ok)
  % mafft --maxiterate 1000 --genafpair  in > out (% einsi in > out)
  % mafft --maxiterate 1000 --globalpair in > out (% ginsi in > out)

If unsure which option to use:
  % mafft --auto in > out

--op # :         Gap opening penalty, default: 1.53
--ep # :         Offset (works like gap extension penalty), default: 0.0
--maxiterate # : Maximum number of iterative refinement, default: 0
--clustalout :   Output: clustal format, default: fasta
--reorder :      Outorder: aligned, default: input order
--quiet :        Do not report progress
--thread # :     Number of threads (if unsure, --thread -1)
```

```
cp /home/yyin/work/class/test-query.fa.cowrument.out.m9.head10.fa .
```

```
mafft --auto  test-query.fa.cowrument.out.m9.head10.fa > test-
query.fa.cowrument.out.m9.head10.fa.l
```

Phylogeny building: FastTree program
(http://www.microbesonline.org/fasttree/)

```
/home/mrupani/Downloads/FastTree
 FastTree protein_alignment > tree
```

```
/home/mrupani/Downloads/FastTree test-
query.fa.cowrument.out.m9.head10.fa.l > test-
query.fa.cowrument.out.m9.head10.fa.l.fasttree.nwk
```

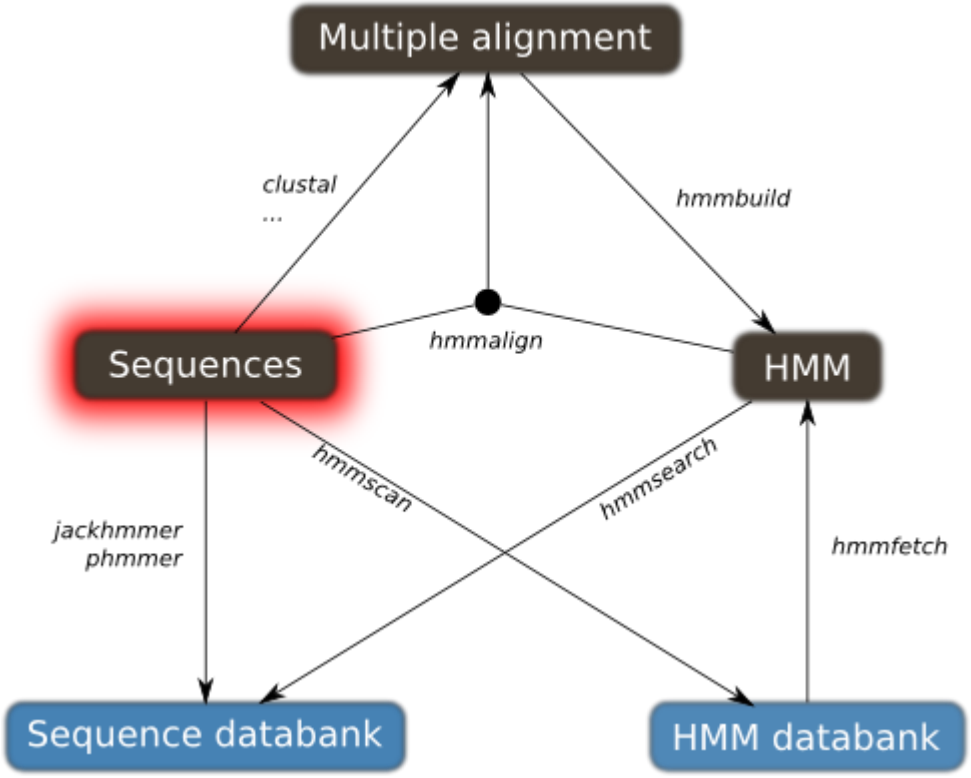
```
less test-query.fa.cowrument.out.m9.head10.fa.l.fasttree.nwk
```

HMMER: http://hmmer.janelia.org/

What is HMMER? ftp://selab.janelia.org/pub/software/hmmer3/3.0/Userguide.pdf

HMMER is a software package that is used for searching sequence databases for homologs, making protein sequence alignments, and making **profile hidden Markov models** (profile HMMs)**.** It implements methods using probabilistic models called **profile hidden Markov models**, mathematically representing multiple sequence alignments.

Compared to BLAST, FASTA, and other sequence alignment and database search tools based on older scoring methodology, HMMER aims to be significantly *more* accurate and *more* able to detect remote homologs because of the strength of its underlying mathematical models. In the past, this strength came at significant computational expense, but in the new HMMER3 project, HMMER is now essentially **as fast as** BLAST



http://drmotifs.genouest.org/2010/10/sequence-hammering/

Go to http://cys.bios.niu.edu/dbCAN/family.php?ID=GH5 and download

```
wget -q http://cys.bios.niu.edu/dbCAN/data/aln/cazy-family/aln/GH5.aln
less GH5.aln

hmmbuild # list options
hmmbuild -h  # list complete options
hmmbuild --informat afa GH5.hmm GH5.aln # build model, afa: aligned fasta format, see User Guide page 16 footnote
less GH5.hmm # profile HMM file is a text file

hmmsearch
hmmsearch -h
hmmsearch --domtblout GH5.hmm.cowrumen.dm GH5.hmm metagenemark_predictions.faa > GH5.hmm.cowrumen.out & # save easy-to-parse table of per-domain hits to file in addition to the regular output (with alignment)
```

```
#
# target name                              accession   tlen query name             accession
#                 1                                        3
#------------------------------------------ ---------- ----- -------------------- ----------
NODE_457020_length_97146_cov_14.955994_orf_01700   -  2    782 GH5.hmm   4          -    5
NODE_457020_length_97146_cov_14.955994_orf_01700   -       782 GH5.hmm              -
NODE_2854003_length_94157_cov_5.769428_orf_67030   -       378 GH5.hmm              -
NODE_2314521_length_30819_cov_0.660826_orf_30190   -       715 GH5.hmm              -
NODE_2314521_length_30819_cov_0.660826_orf_30190   -       715 GH5.hmm              -
NODE_3609387_length_51250_cov_2.036859_orf_24440   -       423 GH5.hmm              -
NODE_2891766_length_19360_cov_5.591064_orf_12550   -       409 GH5.hmm              -
NODE_457020_length_97146_cov_14.955994_orf_01790   -       995 GH5.hmm              -
NODE_457020_length_97146_cov_14.955994_orf_01790   -       995 GH5.hmm              -
NODE_4002281_length_100204_cov_2.154804_orf_16350  -       624 GH5.hmm              -
NODE_421339_length_112723_cov_3.569067_orf_68070   -       413 GH5.hmm              -
```

```
6    --- full sequence --- -------------- this domain -------------   hmm coord   ali coord   env coord
qlen   E-value  score  bias   #  of  c-Evalue  i-Evalue  score  bias  from    to  from    to  from    to  acc description of target
        7       8      9    10  11     12        13      14     15     16    17    18    19
----- --------- ------ ----- --- --- --------- --------- ------ ----- ----- ----- ----- ----- ----- ----- ---- ---------------------
  275   2.9e-71  247.3  13.4   1   2   1.2e-45   8.1e-43  154.0   4.7     2   239    68   328    67   341 0.80 complement(17022..19367)
  275   2.9e-71  247.3  13.4   2   2   2.3e-28   1.5e-25   97.4   0.3     7   228   409   651   403   666 0.74 complement(17022..19367)
  275   2.2e-55  195.2   2.8   1   1   4.6e-58     3e-55  194.8   1.9    22   241    10   271     3   294 0.80 complement(3376..4509)
  275   3.3e-55  194.6   8.6   1   2   4.7e-32   3.1e-29  109.5   1.3     4   243    41   301    38   311 0.80 complement(21709..23853)
  275   3.3e-55  194.6   8.6   2   2   6.9e-26   4.5e-23   89.3   0.4     2   239   344   601   343   628 0.79 complement(21709..23853)
  275   6.2e-55  193.8   3.0   1   1   1.3e-57   8.8e-55  193.3   2.1    24   244    95   357    83   379 0.80 complement(33514..34782)
  275   1.4e-54  192.6   1.1   1   1   2.8e-57   1.8e-54  192.2   0.8    22   242    80   343    73   364 0.81 complement(11478..12704)
  275   1.7e-54  192.3   5.4   1   2   6.3e-29   4.1e-26   99.2   0.6     2   237    41   311    40   322 0.74 34656..37640
  275   1.7e-54  192.3   5.4   2   2   1.1e-27     7e-25   95.2   0.2     2   240   358   625   357   642 0.74 34656..37640
```

[a little parsing, alignment in GH5.hmm.cowrumen.out]

```
less GH5.hmm.cowrumen.dm | grep -v '^#' | awk '{print
$1,$3,$6,$7,$12,$13,$16,$17,$18,$19}' | less
less GH5.hmm.cowrumen.dm | grep -v '^#' | awk '{print
$1,$3,$6,$7,$12,$13,$16,$17,$18,$19}' | awk '$6<1e-2&&($8-$7)/$3>.8' |
sed 's/ /\t/g' | less
```

Extracting domain regions is easy if using perl and bioperl

emboss

```
seqret –help
```
http://emboss.sourceforge.net/apps/release/6.1/emboss/apps/seqret.html

```
seqret -sequence test-query.fa.cowrument.out.m9.head10.fa.l -outseq test-query.fa.cowrument.out.m9.head10.fa.l.aln -sformat fasta -osformat aln
```

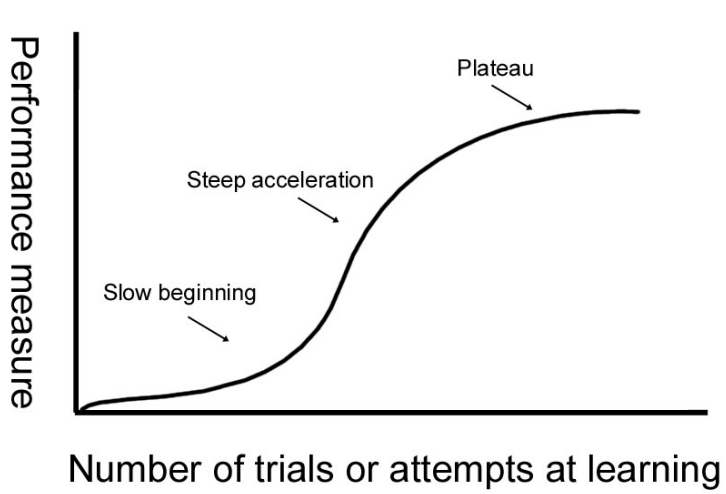
```
infoseq –help
```
http://emboss.sourceforge.net/apps/release/6.2/emboss/apps/infoseq.html

```
infoseq -sequence test-query.fa.cowrument.out.m9.head10.fa -name -only –length
```

More command examples:

```
needle –help
water –help
fuzznuc –help
pepstats -help
pepinfo –help
plotorf  -help
transeq  -help
garnier –help
prettyseq –help
est2genome -help
```

In the remaining classes



Do expect you:

- Get Familiarized with Linux commands
- Be able to read some example Perl scripts
- Know how to run given perl scripts
- Practice examples on projects
- Be able to finish the two course projects

Do not expect you (or not all of you):

- Be able to write complex Perl scripts
- Become a professional programmer
- Become a professional bioinformatian

Things you should know about programming

Learning programming has to go through the hands-on practice, a lot of practice

Hearing what I describe about a command or a program helps, but you will not be able to do it unless you type in the codes and run it to see what happens

Reading others' codes helps but often is harder than writing it by yourself from scratch

Although painful and frustrating, trouble-shooting is normal and part of the learning experience (ask experienced people or google)

To avoid errors, you have to follow rules; most errors occurred in programming are because of not knowing rules or forgetting rules

Use comments in case you forget what you've written means

Edit -> run -> errors -> revise -> errors -> …….. -> run -> success

Good news: finished scripts could be reused or edited for later use

What we will cover in the remaining classes:

Perl basic concepts
Example Perl scripts
Bioperl concepts
Example Bioperl scripts