Isoelectric point histograms and publication quality trees

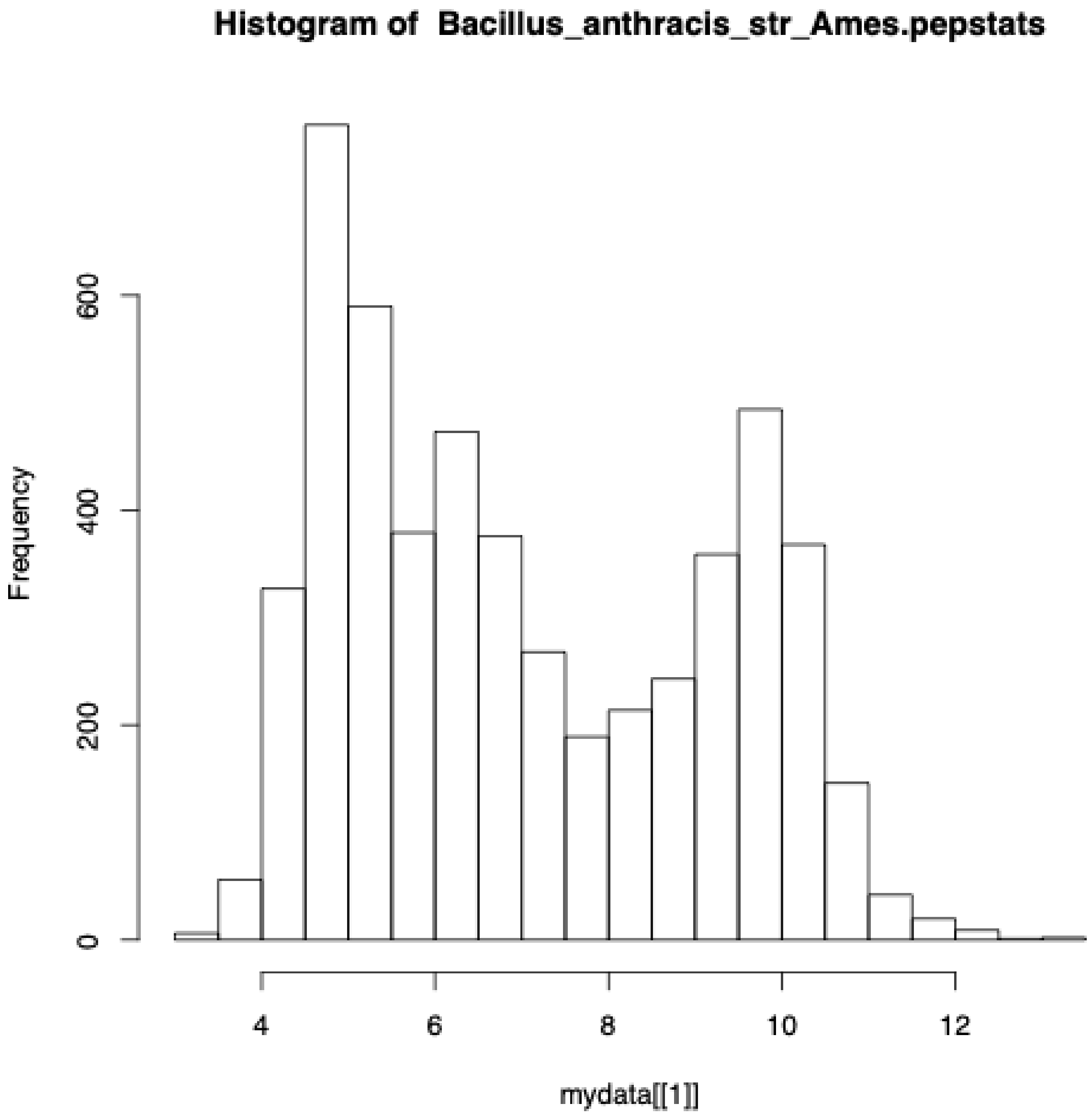
Your name:
Your email address:

1) Histograms of isoelectric points (IEPs) of all proteins encoded in a genome (aka finding organisms that follow a salt-in strategy)

See these the slides for some background.

Underlying today's exercise are two observations:
1) some organisms that live in environments with very high salt concentrations follow a salt-in strategy. Instead of keeping salt out of the cell, they accumulate very high concentration of KCl (>4M salt). A consequence of the high salt concentration is that the reach of a charge is very short (the so-called Debye lengths --- using the formula given here, the Debye length is in the Angstrom range inside the cytoplasm of an organism following the salt in strategy). To compensate for this, organisms following the salt-in strategy have many negatively charged amino acids in their proteins (or in most of them). And these negatively charged aa (sidechains containing a COO- group) lead to an isoelectric point for the proteins at a low (acidic) pH. (I.e. one would need to move the pH of the solution to around pH 2 to have the overall protein with zero charge.)

Aside: most cytoplasmic proteins have a negative charge at the pH at which they normally exist. The overall negative charge prevents proteins from clumping together. Exceptions are proteins that bind to the DNA (the backbone contains phosphate groups that give the DNA on overall negative charge; for a protein to bind to the DNA, it needs to have a positive charge), and proteins that bind to the cell-wall or extra-cellular matrix. The cell wall has an overall negative charge, and for a protein to stay inside the cell wall it helps to have a net positive charge.
For a "normal" cell the theoretical isoelectric points (calculated from the types of sidechains present in the protein) looks as follows:



2) The Haloarchaea follow the salt-in strategy, and as a consequence have one big peak in the histogram of IEPs at below pH 4. The same is true for a group of archaea known as Nanohaloarchaea. The placement of the Nanohaloarchaea remains uncertain. They were initially considered the sister group to the Haloarchaea. Later they were grouped with other recently discovered Archaea into the DPANN group (one of the Ns stands for Nanohaloarchaea). The DPANN group allegedly is a deep branching group in the archaea.
And more recently paper found them to have evolved independently from the Haloarchaea from a methanogen ancestor. A recent article from the Gogarten lab on the origin of Haloarchaea and other halophilic archaea is here. In today's exercise we

will study the IEP profiles for proteoms from Haloarchaea and groups that are possibly related to the haloarchaea.

Nanohaloarchaea.
are a group of small archaea that live in close association with Haloarchaea (ectosymbionts) In 16S rRNA and ATPase phylogenies they often are recovered with the Haloarchaea

Hikarchaeia .
The Hikarchaeia are a recently described group (from MAGs) that likely are the sistergroup of the Haloarchaea. A paper describing them is here (Note the genomes the authors submitted do not include any ORFs - the one file attached below were generated you Yutian Feng from the Gogarten Lab)

Methanonatronarchaeia.
A new group of methanogenes recently described, The placement of this group inside the Archaea remains controversial (here, here and here).

Other groups of possible interest are the Marine Group I Marine Group II and Marine Group III archaea (Ca. Poseidoniales ord. nov.), marine archaea (related to the Thermoplasmatales).

Your task is to determine, if any of these groups contain species that appear to be on the path towards a salt-in strategy.

Every student should analyze at least **one haloarchaeal** (the group is still called Halobacteria at the NCBI), **one nanohaloarchaeal**, and several genomes from each of the proposed ancestral groups.
A selection of multiple fasta files for different genome is here.
You are encouraged to download additional genomes ☺!

Optional:

Go to the NCBI's current genome list.

Click on the "Prokaryotes" tab. Click on Filters in the upper right corner, Check Assembley level "Complete", "Chromosome", and "Scaffold".

Use the use the "Search by organism" box to narrow to a taxonomic group. (Note that after a "Search by organism", one might need to repeat the process of clicking on the "Prokaryotes" tab, and re-tick the filters genomes box.)

Then look for the **R and G** links in the far right-hand column (you will probably have to scroll to the right). The R takes you to a listing of all the refseq files for a genome project (R is referred over G). If you select an organism for which only the G link is available, and if this link does not include an faa.gz file, select a different strain.

Safari no longer connects to FTP servers. Shift or right click on the G or R, copy the link, then go to finder and under the "Go" menu select "Connect to server". Paste the link address, and select "connect as guest".

You want to download the file ending in ".faa.gz". This "faa" file contains all of the proteins coded by a genome.

If this does not work, find the genome of interest and use the genome interface at NCBI, find the genome of interest (e.g., *Dunaliella salina*), place a checkmark into the first column, then select download, the select "download package", then select protein(fasta). NOTE: this does not work on Safari, but works in Chrome!

Download the file to your computer, uncompress the file, and rename it using the Genus_species_strain designation.
Remember **not** to use spaces or special characters in the name!

Using filezilla (**`transfer.cam.uchc.edu`**, username mcb3421usrXX), create a directory for lab13, and transfer the faa files into this directory.

End Optional

We do not need to restrict ourselves to completely sequenced genomes! Feel free to use any other genome you are interested in (halophilic bacteria, acidophiles, human, yeast, ...
Also, the modified version of the script analyzes all faa files present in a directory automatically; therefore, feel free to download as many faa files (the only additional work is to rename the files, so you easily recognize which profile is from which organism)!

Use filezilla to connect to your account on Xanadu.

under **protocol** select **SFTP**
under **host** enter **transfer.cam.uchc.edu**
under **LogOn** Type select **normal**
under **User** enter **mcb3421usrXX** (Go to the class notebook, behind the name of your personal section is an _#. this is your number. If your number is 1, do not use a leading 0. Your username would be mcb3421usr1.)
under **Password** enter *your **new** xanadu password* *(the first password expired; you need to*

use ssh to reset it!)
Hit connect.

Create a lab13 directory, transfer the appropriately named *.faa file into the lab13 directory.

EMBOSS is installed on the cluster. Here is a list of programs in EMBOSS. Today we will be using *pepstats*. Click on its entry in the list to see the command line arguments.

Which organisms did you select? Why are these interesting?

Connect to xanadu in terminal or putty:

```
ssh mcb3421usrXX@xanadu-submit-ext.cam.uchc.edu
```

login – with your new password!

```
srun --pty -p mcbstudent --qos=mcbstudent --mem=2G bash
```

```
mkdir lab13
cd lab13
```

If you did not unzip the faa files on your computer, do this now:
`gunzip *.gz` (the .gz suffix means this text file is compressed, so uncompress it)

`more` *`the_name_of_one_of_your_genomes`*`.faa` (inspect the first few lines of the faa file, type "q" to exit) (...and space to go forward) (...and "b" to go back)

Today we will be using programs from the emboss package, and R and Perl scripts. Thus we need to load the corresponding modules:

```
module load R/4.0.3
module load emboss/6.6.0
module load perl
```

(To check the modules available: `module avail`)

We will use the following scripts today

run_pepstats.pl
parse_pepstats.pl
parse_pepstats_mod2.pl
histogramScript_pdf.R

These scripts are available in this archive
Move them into the lab13 directory using filezilla
(alternative, ssh and cd to the lab 13 directory in terminal and
curl -O https://j.p.gogarten.uconn.edu/mcb3421_2023/labs/lab13/scripts.zip)

finally, here is the pepstats command:

```
pepstats the_name_of_one_of_your_genomes.faa -outfile the_name_of_one_of_your_genomes.pepstats

more the_name_of_one_of_your_genomes.pepstats
```

(inspect the pepstats file, type space to go forward)
(...and "b" to go back)
(...and "q" to exit)

Now we need a program to extract the isoelectric points (amongst other stuff). It's called parse_pepstats.pl. It will work provided the output of pepstats is in a file ending in ".pepstats" (remember that is what you named it above, type "ls" to confirm). Read through the parse_pepstats.pl script. Try to figure out how the program finds the values for the theoretical isoelectric points in the pepstats output.

```
perl parse_pepstats.pl (run the script, and extract the isoelectric points)
ls -l (you made a bunch of additional files)
head the_name_of_one_of_your_genomes.pepstats.pI (the first 10 isoelectic points!)
head the_name_of_one_of_your_genomes.pepstats.parsed
```

(the columns are the accession number of the protein, the length of the protein, the theoretical isoelectric point, and fraction of positively charged residues)

Use filezilla and drag the file from the lab13 folder containing the isoelectric points (.pepstats and .pI ending) and the table with the parsed output (ending on .parsed) to your computer.
Load the .parsed file into Excel.

You might want to skip the stuff in blue, and rather use the automated procedure below: Make histograms of the pI data in Excel, (remember to select "All Files" to see the file in the Excel load window), use Insert -- Statistic Chart (all-blue column chart icon in the Charts section) -- Histogram.

Select a few proteins with very alkaline theoretical IEP, copy their accession number, and then use Entrez to determine the function of these proteins. (see the questions below).

This is a rather tedious procedure that can be easily automated:

`run_pepstats.pl` (is a script that runs pepstats on all faa files in the directory, use nano or more to inspect the script). To run
`perl run_pepstats.pl`

`parse_pepstats_mod2.pl` (runs parse_pepstats on all pepstat output files, reformats the .pI file and hands it over to an R script that makes histograms, and finally, renames the histograms).
`perl parse_pepstats_mod2.pl`

Briefly read through the scripts (they should be already in the lab13 directory) to understand what they do.

`ls`	(to make sure the scripts are in your directory)
`perl run_pepstats.pl`	(runs pepstat on all *.faa files in the lab9 directory)
`ls`	(to check which files were created)
`perl parse_pepstats_mod2.pl`	(run parse pepstats on all files and extract the isoelectric points and creates a histogram)
`ls -l`	(you made a bunch of additional files)
`head G_species.pepstats.pI`	(the first 10 isoelectic points!)
`head G_species.pepstats.parsed`	(check the table that summarizes the results for each protein)

For each pepstats file a pdf file containing the histogram should be created in the folder.

Use filezilla to drag the file(s) from the lab13 folder containing the isoelectric points (.pepstats.pI ending) and the table with the parsed output (ending on .parsed) and the .pdf files with the histograms to your computer.

For at least three of the analyzed genomes, describe the distribution of isoelectric points.

How many peaks?

Why might there be a minimum at around pH7.5?

Compare your finding with others in the class.

Check a few of the ORFs with very alkaline theoretical isoelectric point (the *.parsed outfile contains accession numbers in the first column, sort on the IEP (in the 3rd column), using entrez Protein with the accession number will get you to the genbank entry for that protein (sorry, this does not work for the Hikarchaeia); if you seem to be stuck with hypothetical proteins, pick proteins that are longer).
What functions do these genes have?

Which charge would these proteins have at neutral pH? Can you see a pattern in the types of enzymes?

Place the histograms of the theoretical isoelectric points into your notebook.

How to learn R:

The command line and scripting languages can make your life a lot easier!
Work that is impossible using a GUI, is easily accomplished using simple scripts (shell scripts, python, perl) that execute system commands (via the command line) repetitively.
Regular expressions allow to extract information from large files (e.g., species names from comment lines; details of protein statistics).
A large genomics and bioinformatics community exists that uses the R programming language for statistical computing.
John Hopkins University is offering a course on R (not focused on Bioinformatics) via coursera (the course is free, you only pay, if you want a certificate). The course uses Rstudio, which provides an excellent notebook environment to keep track of your projects.
Software carpentry workshops offered at UConn provided intros to the command line, R!, and github, but the instructors moved on to other jobs … Workshops through statconsulting were useful, but none are currently in planning https://statsconsulting.uconn.edu/workshop-schedule/

Courses at UConn that provide an intro to R:
https://innovatelabs.uconn.edu/tech/r-programming/
ECON 3321 https://catalog.uconn.edu/directory-of-courses/course/ECON/3321/

GRAD 5100: Fundamentals of Data Science
Open only to students in the Data Science M.S. program.
Adam Zweifach teaches a course next semester on “Design, statistical analysis and presentation of biological laboratory experiments.” It will also introduce R and R Studio.

2) Tree Images in Figtree

Figtree is a very useful program to create publication quality trees. Is also can save them as vector graphics, which makes it easy to edit them in Inkscape or Adobe Illustrator. See slides from Monday’s class.

1) Check if figtree is installed on your computer. If not, download and install figtree from the github page

2) The trees we will look at today were calculated for exteins and inteins of the terminase gene in the Actinophages from subcluster C1. The nucleotide sequences were retrieved from phagesDB, renamed to include the state where the phage was isolated, aligned in seaview using muscle, and separate datasets created for intein and extein sequences. For the latter, three different files were created

- a multiple fasta file containing all terminase exteins from cluster C1
- the same as above, but the very divergent sequence from Toneneli removed
- only the extein sequences from phages that contain an intein
- all intein sequences

Each of these phylogenies were calculated using iqtree2. These phylogenies are available in Newick format in this text file

- Open the text file with the tree in a text editor (BBEdit or Notepad++).
- Copy the intein tree onto the clipboard
- Open figtree
- Click in the central figtree window
- Paste the tree (ctrl-V)

3) It is a good idea to start with the intein tree.
Do the following:

- Select "Node" in the central selection field
- click on the long internal branch, select re-root
- move root length lever to the left (we have no idea where this is rooted.
- place check mark into branch labels (in the menus on the left)

- open branch labels menu (on the left)
- select label in the display pull-down menu (these are the bootstrap support values that iqtree wrote into the newick formated file)

- in the selection indicator on top, select taxa
- click on one of the two basal branches
- select a color for the leaves (aka OTUs)
- click on the other basal branch and select a different color (Red and green or blue and green work well)

As many of the branches have very low support values, we can display the support in a different way

- Open the "Appearance" menu on the left
- under "colour" by select "label"
- check mark into gradient
- click on setup Colours (sic)
- Adjust the hue levers at the bottom to have the spectrum go from yellow to red (or from yellow to blue); move the other three levers to the right
- The well supported branches now are in blue and the branches that do not have high support values are in green and yellow

- You can modify other things like font sizes, placement of the labels, line width, add a legend,   ...

Figtree is intelligent.
The tree we have displayed now was calculated from inteins in terminase genes in genomes of tailed actinophage (viruses that infect actinobacteria) subcluster C1. The terminase is involved in packaging the DNA into the phage heads)
The names are composed of the state (or other indication of from where the phage was isolated). The displayed tree was calculated from the intein only.
The extein sequences have the same names, but obviously the extein tree is expected (and it is) rather different.

Open a new Figtree window (***leave the old tree open in its window***!)

Copy paste one of the extein trees (below) into the empty Figtree window.
There is no harm, if you do this for all three extein trees.

If everything works as expected, the colors selected for the leaves on the intein tree are applied to the new tree.
Under file select new.

Re-root the tree in one of the longer branches (so that it looks balanced). If you use the extein tree with the toneneli sequences you could use this as outgroup. Set the root branch length to zero. "Colour" the branches based on support values (see above).

The clans defined by the intein tree are not reflected in the extein phylogeny. The two intein types were transferred several times between different exteins. Note the nearly identical intein between red and green leaves.

Give two examples for very similar inteins found in divergent exteins clans, i.e., list at least two pairs of actinophages with divergent extein sequences but with similar inteins reflecting recent intein sharing.

Can you find any very similar inteins from a similar geographic distribution, but the exteins are dissimilar?