HORIZONTAL GENE TRANSFER DETECTION

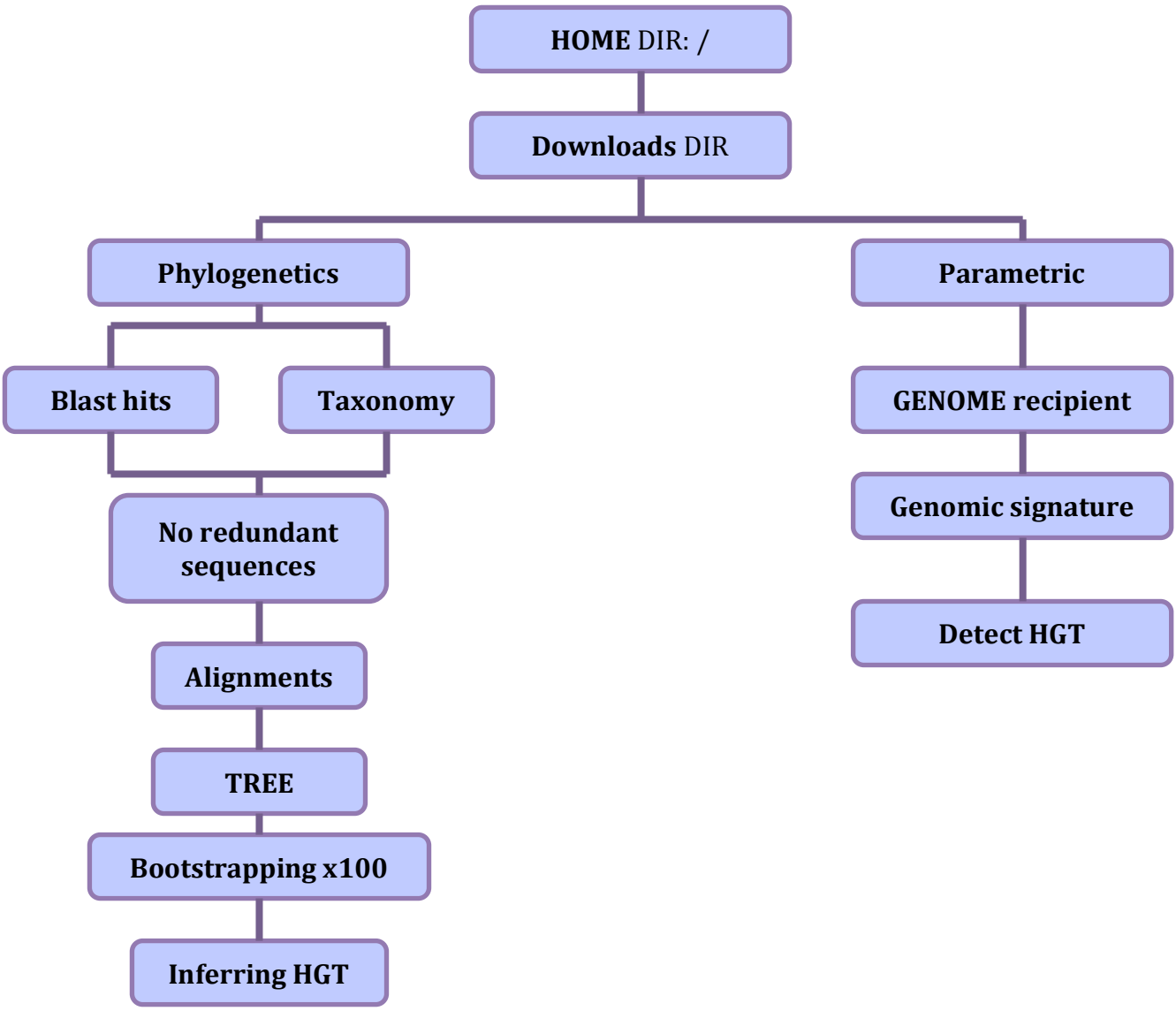
Sequenzanalyse und Genomik
(Modul 10-202-2207)

Alejandro Nabor Lozada-Chávez

Before start, the user must create a new folder or directory (WORKING DIRECTORY) for all data and results produced with this analysis. For example: If your current location is inside of the folder Downloads, make a new directory named as "phylogenetics".

Before start, the user must check the use of the following commands: **grep** (option –c), **cut** (–d –f), **sed** (option -E), **tr**, and **awk** (option $1 and $0).

Finished examples, plots, phylogenetic trees, genomic signatures with parametric methods, and PERL and R scripts can be found in **Dropbox**:
https://www.dropbox.com/sh/64vcu0txj17kdvg/AABr0mHr4VirHvTokxxZvLMUa?dl=0
Permanent link at **GitHub**:
https://github.com/naborlozada/Sequenzanalyse-und-Genomik-Leipzig



/// PHYLOGENETIC APPROACH ///

Resources
BLAST website: https://blast.ncbi.nlm.nih.gov/Blast.cgi

QUERY SEQUENCE:
Complexity Hypothesis revisited:
https://www.ncbi.nlm.nih.gov/protein/159904610

Dead-end HGT event:
https://www.ncbi.nlm.nih.gov/protein/20091267

STEP 1
Finding Possible Homologs/Xenologs

GOAL: Make a BLAST (find putative homologs) using our query sequence.

1) My **QUERY SEQUENCE** must be saved in FASTA FORMAT in a plaint text file. Save it your current working directory.

2) Make a BLAST:
   a. GO to BLAST website, choose **Protein BLAST** (protein ➔ protein)
   b. PASTE you QUERY sequence in "**Enter Query Sequence**".
   c. FILL the options:
      i. In Database ("Choose Search Set"), choose "**Non-redundant protein sequences (nr)**".
      ii. In Algorithm ("Program Selection"), choose "**blastp**".
      iii. GO to "**Algorithm parameters**".
      iv. In **General Parameters** choose:
         1. **Max target sequences** (500 or 1000).
         2. Click-in "**Short queries**" field.
         3. Keep the default **Expected threshold** equal to 10.
      v. In "**Scoring Parameters**":
         1. Matrix: BLOSUM62.
         2. Gap cost: 11/1

3. Compositional adjustments: Conditional compositional score matrix adjustment.

vi. In "**Filtering and Masking**" section:
   1. Filter: Click-in "**low complexity regions**":
   2. Mask: **Mask for lookup table only** and **lower case letters** will be *empty fields*…
   3. Click-in "**Show results in other window**", then…

3) Click BLAST and wait.

4) DOWNLOAD:
   a. The set of sequence output produced by BLAST. HOW-to:
      i. GO to the "**Description**" section, in Select make a click in **All.** Click **Download** and choose only FASTA (complete sequence). Click in **Continue**.
      ii. SAVE this file in your current WORKING DIRECTORY. Sometimes this fasta file is saved and named automatically in the Downloads or in My Documents directory as "seqdump.txt". Move this file (seqdump.txt) to your current working directory and renamed as, for example: `BLAST_hits_results_sequences.fasta`

   b. Hit table information: this file contains basic information of the BLAST results (displayed by columns: query id (Query sequence), subject ids (NCBI database), %identity, %positives, alignment length, mismatches, gap opens, q. start, q. end, s. start, s. end, evalue, bit scores.
      i. GO to the "**Description**" section, in Select make a click in **All.** Click **Download** and choose only Hit Table (text). Click in **Continue**.
      ii. SAVE this file in your current WORKING DIRECTORY. Sometimes this text file is saved and named automatically in the Downloads or in My Documents directory with the automatic name (for example: 5B6HWAC001R-Alignment.txt). Move this file to your current working directory and renamed For example: `BLAST_hits_results_info.txt`

STEP 2
Rename Sequence Fasta headers and obtain the Taxonomic Information

GOAL: Replace the long fasta sequence annotation with a GI number (only 10 characters) for all sequences in the file.
From this:

```
>WP_012193024.1 30S ribosomal protein S2 [Methanococcus maripaludis] QWEMMX
A9A7B6.1 RecName: Full=30S ribosomal protein S2
```

to this:

```
>501144881
```

Working in your directory "phylogenetics":

5) Create a new FASTA file with the new short FASTA headers (keep the **Accession number**).
   a. HOW-to: Open a terminal (go to your directory) and type the following syntaxis in a single line (separated by spaces). Within the double quote (" ") must be *one single space*:

```
cut -d " " -f1 BLAST_hits.fasta > BLAST_hits_short.fasta
```

6) From the new FASTA headers create a list of all ACCESSION numbers.
   a. This list will be separated by one space or tabular spaces, and the '>' symbol must be remove.
   b. Important: The **order** of these accessions numbers should be the same than in the FASTA file, from the top to the bottom.
   c. HOW-to: Open a terminal (go to your directory) and type the following syntaxis in a single line (separated by spaces):

```
grep ">" BLAST_hits_short.fasta | sed -E 's/>//' | tr "\n"   " "
> myACCESSION_list.txt
```

This combination of commands is called **pipe-line** (pipe symbol "|").
Save the output in a text file (**myACCESSION_list.txt**) or copy it, to paste it later in a specific line in the **get_taxa.sh** script. READ instruction in the script before doing this.

7) Get TAXONOMY information from the Gene Accession numbers:
   a. HOW-to: OPEN the bash script **get_taxa.sh** and read the usage instructions.

8) Keep FASTA headers ≤10 characters. Replace the current FASTA headers using the perl script: **`change_fasta_headers.pl`**. Read the usage instructions before used it.
   a. HOW-to: Open a terminal (go to your directory "phylogenetics") and type the following syntaxis in a single line (separated by spaces):

```
perl change_fasta_headers.pl  >  BLAST_hits_short.GI_headers.fasta
```

HINT: You need a table with ≥ 4 columns separated by Tabular spaces. Importantly, the first must represent the same ACCESSION number contained in your fasta file (XP_011328631.1), while the second column is the GI number (758212931) obtained from the taxonomy info. These two columns are key for the script. Save this table with and specific name, such as "my_TAXA_table.txt". Important, in the code of the script **`change_fasta_headers.pl,`** replace the names of the corresponding "fasta sequence" and "taxonomy file", respectively.

An illustrative format of the table in excel format (*.xls) can be found in the folder in "phylogenetics" in the DropBox section.

STEP 3
Reducing Redundancy, Sequence Alignment and Phylogenetic Reconstructions

GOAL: Exclude all highly similar sequences and retain in the fasta file the remaining sequences. Next, obtain an alignment and reconstruct a tree.

9) REDUNDANCY: Reduce redundancy at 90% of similarity.
   a. HOW-to: Open a terminal (go to your directory "phylogenetics") and type the following syntaxis in a single line (separated by spaces):

```
cd-hit
-i BLAST_hits_short.GI_headers.fasta
-o BLAST_hits_short.GI_headers.nr90.fasta
-c 0.90
```

Important: For the alignments and future sequence analysis only use the file with the extension "*.fasta" (the file from the –o option). The other file produced by cd-hit (*.fasta.clstr) contain "clustering information" and any sequences are not contained there. Also, take in account that your query sequence should be deleting in this step. Check it, if it was deleted just added from the original query sequence (NCBI or file).

While doing this step you must wonder, "How much redundancy should I retain/delete?" Here, think in gene copies or genes from highly similar strains.

10) ALIGNMENTS: Make a multiple sequence alignment of your FASTA file.
   a. HOW-to: Open a terminal (go to your directory "phylogenetics") and type the following syntaxis in a single line (separated by spaces):

```
mafft BLAST_hits_short.GI_headers.nr90.fasta   >
BLAST_hits_short.GI_headers.nr90.aln.fasta
```

Improve your alignment using MAFFT with other options, such as 'retree', 'einsi', maxiterate' and others. Check the manual document.

11) Selection of best-fit model of protein evolution using PROTTEST. This step takes too much computational power and time. Therefore, you must do it when you have both. **Skip this step for now and go to the "Change sequence format" step.** For this and other exercises in this course you don't need it, instead, we will use a conservative substitution model for your sequence alignment in RAxML (i.e.

PROTGAMMA+WAG model). Despite this, the command line to execute this task is shown below for your interest:

a. HOW-to: Open a terminal (go to your directory "phylogenetics") and type the following syntaxis in a single line (separated by spaces):

```
java -jar /MY/FULL/PATH/DIRECTORY/prottest3-master/dist/prottest-
3.4.2.jar
-i BLAST_hits_short.GI_headers.nr90.aln.fasta
-all-distributions
-F
-AIC
-tc 0.5
-threads 4
-o BLAST_hits_short.GI_headers.nr90.aln.prottest_output.txt
>  BLAST_hits_short.GI_headers.nr90.aln.prottest_errors.txt
```

* IF PROTTEST was installed correctly, just call the program by typing: **./prottest**
Make a test:
/MY/FULL/PATH/DIRECTORY/prottest3-master/dist/prottest3 -h

12) Change sequence format: from FASTA format to PHYLIP format. Why? Many (or almost all) programs that reconstruct phylogenetic trees only accept a PHYLIP format.

a. HOW-to: Open a internet browser (firefox, chrome, safari…) and ask for any help to GOOGLE by typing: from **fasta** to **phylip** format:

First hit is a good option:
sequenceconversion.bugaco.com/converter/biology/sequences

I. Convert file from: **fasta**
II. To: **phylip**
III. Alphabet: **protein**
IV. Click-in **Choose file:** upload your fasta sequence
`BLAST_hits_short.GI_headers.nr90.aln.fasta`
V. Click-in **Convert**
VI. Move your phylip file in your working directory "phylogenetics", and rename:
From "sample.phylip" to "BLAST_hits_short.GI_headers.nr90.aln.phy".

HINT: There are several resources offered as script programs or services online. You can ask to google something like: "fasta to phylip format online" or "fasta to phylip format script" to find other ways to solve this issue. WARNING: By using the scripts of others, you must be aware that some scripts could contain some mistakes. Make a double check.

Save the produced new PHYLIP format file in your current WORKING DIRECTORY.

13) Phylogenetic reconstruction using RAxML.
    a. HOW-to: Open a terminal (go to your directory "phylogenetics") and type the following syntaxis in a single line (separated by spaces):

```
raxmlHPC
-m PROTGAMMAWAG
-s BLAST_hits_short.GI_headers.nr90.aln.fasta
-n BLAST_hits_short.GI_headers.nr90.aln.raxml
-p 12345
```

14) OPEN YOUR TREE. RAxML produces 5 files, but your final tree is in the file designed as "RAxML_result.*": **RAxML_result.my_tree_name.raxml**

    a. HOW-to: GO to GOOGLE and ask for help: open phylogenetic tree online. Use the first option named as "Tree - Viewer" from the ETETOOLKIT group (http://etetoolkit.org/treeview/).
        i. Copy the newick tree format of your tree (the text format) and paste it in the option "Paste your tree in newick format". Also, you can visualize your alignment; just paste the alignment in fasta format.
        ii. Observe the branches. Can you find your query sequence?

    ADDITIONAL info: The other files produced by RAxML contain a summary of the job (RAxML_info.*), errors information during the job (RAxML_log.*), a best tree found by parsimony (RAxML_parsimonyTree.*), and a preliminary best tree (RAxML_bestTree.*). Read the manual of the RAxML program to know more.

15) RENAME LABELS: in order to observe the taxonomic distribution of each gene across the tree, the current branch labels of the tree (GI_NUMBERS) must change.
    a. HOW-to: Make a table of **two columns**: GI_NUMBER vs FULL_TAXONOMY. Save it as "GInumber_to_TaxaInfo.txt".
    b. USE Newick Utilities program to rename the tree: HOW-to: Open a terminal (go to your directory "phylogenetics") and type the following syntaxis (separated by spaces):

```
nw_rename
BLAST_hits_short.GI_headers.nr90.aln.tree  GInumber_to_TaxaInfo.txt  >
My_NewTree_with_Taxa.tree
```

    HINT: An illustrative table of two colums (*.LABELS.txt) and examples for renamed trees (figures in PDF format) can be found in the folder in "phylogenetics" in the DropBox section.

16) OPEN THE TREE: Use online servers or Download FIGTREE:
    a. HOW-to: GO to the FigTree website (http://tree.bio.ed.ac.uk/software/figtree/)
    b. Download the program for linux named as "FigTree_v1.4.3.tgz".
    c. Decompress the file.
    d. Open a terminal (go to your directory "phylogenetics") and go to following directory of FigTree:

```
cd /scratch/u/home/praktikum/Downloads/FigTree_v1.4.3/bin/
```

    OR starting from "Downloads":

```
cd /FigTree_v1.4.3/bin/
```

    e. Type only: **figtree &**

17) MANIPULATE THE TREE: Open the tree and observe the taxa distribution with your new branch annotation. Play a bit with the options of the program and find the best way to illustrate the HGT event.
    a. Bottoms such as "Node, Clade, Taxa" at Selection mode, "collapse branches", "color", "highlight", and "find" can be useful.

STATISTICAL SUPPORT FOR TREES:
The Non-Parametric Bootstrap & Testing
The Reliability of a Tree Topology

Bootstrapping is the way of testing the reliability of a tree topology produced from a dataset (e.g. sequences). From your sequence alignment:
How many times out of 100 (can be also 1,000, or 10,000…), the same branch was observed when a phylogenetic tree is reconstructed on a resampled set of sequences?

A) FAST-BOOTSTRAPPING & Maximum Likelihood search using RAxML.

a. HOW-to: Open a terminal (go to your directory "phylogenetics") and type the following syntaxis in a single line (separated by spaces):

```
raxmlHPC-PTHREADS
-f a
-m PROTGAMMAWAG
-p 12345
-x 12345
-# 100
-s BLAST_hits_short.GI_headers.nr90.aln.phy
-n myTREE_w_100_BOOTSTRAPS.raxml.tree
-T 4
```

Where, 'a' represents the call to *rapid bootstrapping* option, and '#' the number of times your data (alignment) will be re-sampled and the number of trees that will be reconstructed.

WARNING: this method requires too much computational power and time. Try to estimate the time in which your tree will be finish. Take in account the following: 1) A ML tree (171 aligned sequences and 467 sites) was reconstructed in `739.200006 secs` (21 min.), 2) a bootstrapped tree using the same alignment with 100 replicates was reconstructed in `21640.823654 secs` (6 hrs). According to that, then, how much time your tree will need approximately to be reconstructed?

The minimum number of replicates accepted in a bootstrapped tree is 100, while the minimum best number of replicates is 1,000. For simplicity, a number of replicates around 20 will be fine if you have a long (number f sites) and large (number of sequences) alignment.

B) Obtain a CONSENSUS TREE using CONSENSE.

a. HOW-to: Open a terminal (go to your directory "phylogenetics") and call the following program and type enter:

```
consense
Please enter a new file name> myTREE_w_100_BOOTSTRAPS.raxml.tree
Y   [Type 'Y' if you are agree with all the options, and press enter]
```

Consensus, a program from the phylip package, produces two files, **outfile** and **outtree**. The first one contains all information of the resampling and ML trees calculations to produce the *consense tree*, which is based on the 'majority rule' (MR). This method can be changed, check letter C at the menu in consense. The second file contains the consensus tree with the bootstrap percentage scores.

Rename the tree and open it in FigTree. In Figtree, click-in the option Branches labels to show the bootstrap values at each node.

WARNING: Although there are other methods to reconstruct the consensus tree, e.g. RAxML, the most popular and easy form to get this consensus tree is using consense from Phylip.

C) RENAME and MANIPULATE THE TREE for the HGT analysis.

a. HOW-to: Repeat all previous steps (from 15-to-17) to rename and visualize the tree with a proper annotation.

WARNING: Probably the order of the branches in the final bootstrapped tree is not the same than the ML tree reconstructed previously (step 13). So that, be careful when you rename the tree.

D) Compare the trees and show bootstrap values:

a. Open the final reconstructed and annotated tree from the step 13, and open the bootstrapped tree. In the latest, click-in the "Branch labels" options and observed the bootstrapped values.

/// PARAMETRIC APPROACH ///

CAI vs GC content

GOAL: Identify potential HGT events through the genomic signature of a reference transferred gene (the query sequence).

Here, the user needs to create a directory named "parametrics". For example, if your "phylogenetics" directory is located within DOWNLOAD directory, make the "parametrics" there. Therefore you will have two working directories, the previous "phylogenetics" and the current "parametrics". All analysis produced in this section should be done in "parametrics".

Working example:
Recent HGT event in E. coli genome:
ftp://ftp.ncbi.nlm.nih.gov/genomes/genbank/bacteria/Escherichia_coli/latest_assembly_versions/GCA_000005845.2_ASM584v2

YOUR GENOMES:
ftp://ftp.ncbi.nlm.nih.gov/genomes/refseq/

Additional sources:
https://www.ncbi.nlm.nih.gov/genome/
https://www.ncbi.nlm.nih.gov/Taxonomy/Browser/wwwtax.cgi

1) DOWNLOAD THE GENOME of your QUERY sequence organism.
    a. **HOW-to:** GO TO: ftp://ftp.ncbi.nlm.nih.gov/genomes/refseq/
    b. Choose the group of the QUERY organism: Plant, Fungi or Bacteria.
    c. Find the name of the organism. TIP: Ctrl-F and type the name (e.g. Physcomitrella_patens).
    d. GO to "latest_assembly_versions" directory and
    e. SAVED only the file with the extension "*_cds_from_genomic.fna.gz" in your CURRENT WORKING directory "parametrics". The * can be any name, and the notation **cds** represent **Coding Sequences** in nucleotides.

2) Open this file and rename manually your QUERY SEQUENCE.
    a. HOW-to: Ctrl-F your QUERY sequence annotation (either ACCESSION or GI number). Next, rename your QUERY fasta annotation only either with ACCESSION or GI number.
3) Using CODONW to calculate the Codon Adaptation Index (CAI) and the percentage of GC content (GC) per gene.
    a. HOW-to: Open a terminal, move to your "parametrics" directory, and type the following syntaxis as in the example (separated by spaces):

```
codonW -h

codonW [inputfile] [outputfile] [bulkoutfile] [options]

example:

codonw my_genome.CDS.fna my_genome.CDS.out my_genome.CDS.blk
-all_indices -coa_cu -coa_rscu -rscu –cu -cutab -nomenu –nowarn –silent
```

HINT: For statistical support and another analysis, have a look in the options: "-coa_cu", "-coa_rscu", "-rscu" "–cu", and "-cutab".

4) Next, make your own gene annotation.
    a. HOW-to: Open in a TEXT editor the file my_genome.CDS.out.
    b. Copy-and-paste all content in a excel sheet page and add new column with the title CLASS.
    c. FIND your QUERY sequence and annotate as HGT in the CLASS column. The rest of the sequences should be annotated as HOST.
    d. Copy this new table in a text file and save it as "my_genome.genomic_features.CDS.out.txt".

5) PLOT the genomic signature.
    a. HOW-to: Open a terminal in your current working directory and use the R script make_plot.R to plot the result. Read the instruction to use it.
    b. In the terminal, type the following syntaxis (separated by spaces):

```
Rscript make_plot.R  <My_TABLE>  <MY_OUTPUT_FIGURE>  <SAVE_Rjob.Rdata>

Example:
  Rscript make_plot.R my_genome.genomic_features.CDS.out.txt
  my_genome.genomic_features.CDS.figure.pdf
  my_genome.genomic_features.CDS.figure.Rsession.R
```

Type Rscript make_plot.R (then press enter) to see more information.

6) Open the figure and OBSERVE the distribution of the putative HGT candidate among the gene of the recipient genome.