Final project

Genomic Data Analysis and Visualization 2019

Your lab just identified an interesting effect in a hot spring located in Iceland, which, coinciding with regular activities in a nearby volcano, experiments events of high temperature. You noticed that, after such episodes of high temperature (close to 90 degrees!), a bloom of algae living in the same environment happens.

To investigate this effect, you got a very generous funding from Tyrell Corp that will allow you to perform an in depth genomic exploration of this singular hot spring ecosystem.

1. Metagenomics

As a first step, you decide to run shotgun metagenomic sequencing of the prokaryotic microbiome in two of kind of samples: 1) one taken during the high temperature episodes, and 2) right after the episodes, when the temperature is back to normal and there is a bloom of algae.

After months of waiting, the sequencing results from your two metagenomic samples just arrived! The raw read files (reverse and forward) were produced by Illumina pair-end sequencing and are located in your computing server:

```
/home/2019_2020/data/metagenomics/hotspring-hightemp.1.fq.gz
/home/2019_2020/data/metagenomics/hotspring-hightemp.2.fq.gz
(forward and reverse reads from the high temperature sample)

/home/2019_2020/data/metagenomics/hotspring-normaltemp.1.fq.gz
/home/2019_2020/data/metagenomics/hotspring-normaltemp.2.fq.gz
```

`(forward and reverse reads from the `***`normal`***` temperature sample)`

That means work to do!
Investigate the following basic questions:

- What is the most abundant organism in *high-temperature*?
    - What's its relative abundance?
    - How do you interpret the abundance number obtained?
    - Is it a novel or known species?
    - If possible, describe the most important features of such species.

- What's the level of the most abundant organism in normal-temperature?
    - Is the high-abundant species in the high-temp sample detected here?
- Why algae are observed/not-observed in the normal-temperature condition?
- Briefly describe your hypothesis explaining the differences observed between high and normal temp samples.

2. Genomics

"You are very excited with your preliminary findings that one specific organism is very abundant in high-temperature episodes. To further characterize it, the lab isolated it and sequenced cDNA of samples from both normal and high-temperature conditions, two biological replicates each. They performed quality checking, providing us only high quality reads in fasta format.

You sit in front of your computer. Your coffee cup is still smoking and everybody is silently hitting the keyboard in the lab. You open a bash terminal and `ls` the directory where you left ready both reference data and the raw sequencing reads. There they are. First things first… you want to be sure what you are dealing with. You uncompress your data and begin to analyze... "

Untar and uncompress your `data.tar.gz` and **start by checking your samples** trying to answer the following questions:

1. How many samples do you have?
2. How many reads do you have in each of your samples?
3. What kind of reads are they? (e.g. paired-end reads, mate-pair, single-end…)
4. Are all the reads of the same length?
5. Just from the files you have been provided, could you say something about reads orientation (5' to 3', 3' to 5')? And what about DNA strand (forward or reverse strand)?
6. Is there any additional comments you would like to do about your reads?

3. Read mapping

"Before performing other downstream analyses (variant calling, expression analysis, etc) you need to map your reads to the reference."

First, you will need to **create an index of the reference genome** (tip: use the `bowtie2-build` command).

Next, **map each of your samples to the reference genome** using Bowtie2 (http://bowtie-bio.sourceforge.net/bowtie2/). (tip: check the `bowtie2 --help` for a parameter which allows you to use fasta instead of fastq files as input; also make sure to redirect the stderr output of bowtie2 to a file using the `2>` redirection, so that you can collect bowtie2 mapping stats).

1. How many records are in your mapping (.sam/.bam) files? How many different reads are in your mapping (.sam/.bam) files? How these numbers compare with the number of reads in your original samples and with the alignment statistics (stats from bowtie2)?
2. How many reads map to a single location and how many to more than one (multiple mapping reads)? How do you think that multiple mapping reads could affect downstream analyses (variant calling and RNAseq)?
3. Could you use these mappings to perform an analysis of Copy Number Variation (https://en.wikipedia.org/wiki/Copy-number_variation)?

4. Variant calling

"The next step is performing variant calling to compare the DNA in your samples with that of the reference genome. Maybe some mutation is responsible for the sudden proliferation of these organisms..."

To perform variant calling you can use the mappings from all your samples. To do that, **sort the mappings** of each of your samples (tip: you should use `samtools sort` for that). Then, you can **merge the sorted files** into a single BAM file (tip: you should use `samtools merge` for that). Try to do all these steps so that the header information is present in the final merged BAM file (tip: check the parameters available in the help of the different `samtools` commands).

Finally, **perform the variant calling** using `bcftools mpileup` and `bcftools call`. Check the parameters used in the practice lessons and apply those. (tip: remember that you can pipe commands in linux using "|").

Transform, to a "tab" separated file, the BCF file you obtained, including the variant ID, its position, the reference allele, the alternative allele, the variant quality and the depth of coverage of the variant (tip: use `bcftools view calls.bcf | grep -v "^#" | cut -f 1,2,4,5,6,8 | sed 's#DP=\([0-9][0-9]*\).*#\1#' > calls.tsv`. You could run the previous command line with and without the `sed` command to check what is its purpose).

1. How many variants did you obtain? How many are SNPs, how many insertions and how many deletions?
2. How many variants have quality greater or equal than 10?
3. How many variants have depth of coverage greater or equal than 10?
4. Identify the variant with best quality. Could this variant be affecting a gene? (tip: compare the position of the variant with the positions of the genes in the genome GFF file). Which gene did you find, if any? Without actually checking it, could you give an example of how your variant could be affecting a gene product (e.g. a protein)?

5. Repeat the variant calling using only a single .bam file, instead of merging them. What is the main difference you find in the results when you use only one file?
6. (extra question which you may take some time...) Download, to your local machine, the files with the mappings and the fasta file of the genome. Use them to locate in IGV the best variant you have, and capture the image of the variant. Note that you will likely need the indexes of the mappings (.bai) and genome (.fai) files.

5. Differential expression analysis

"Given that you have expression data for each sample, you can also compare the expression differences between the samples grown under normal and high-temperature conditions, which could provide some additional clue of important genes involved."

For differential expression analysis (DEA) you need to start using the sorted bam files generated previously. First of all, you need to do a "read-count" using `htseq-count`. Some important parameters: you have to use ***`-i Name`*** *and* ***`-t gene`***. (tip: you need to use a .gff file to count the reads). Once you have the four count files is necessary to merge all together and save it as `count.txt` file (tip: explore ***`join`*** command). You should finally obtain something like this in your ***count.txt*** file:

```
> cat count.txt
#genes        #counts (The header it's not necessary)
NP_212986.1 226 189 195 221
NP_212987.1 12 9 10 7
NP_212988.1 27 70 25 29
NP_212989.1 97 100 100 68

…                   ….
```

Now you can use your counts to perform the DEA analysis. For that use the Bioconductor package DESeq2, using these loading data parameters:

```
> cat DESeq2.R
# Loading Data in R #

counts = read.table("counts.txt", header=F, row.names=1) # Load the raw counts table
colnames = c("Normal","Normal","High","High") # names for column names
my.design <- data.frame(row.names = colnames( counts ),
                        group = c("Normal","Normal","High","High")
) # our experiment design for DESeq2 analysis
```

And be sure you write properly the contrast analysis in DEF section :

```
#DEF section
res <- results(dds, contrast=c("group","Normal","High"))
```

If you inspect in R the data frame of your experiment design (my.design variable) it has looks like this:

group

V2	Normal
V3	Normal
V4	High
V5	High

Important note: *#rlogtranformation for PCA analysis section of the DESeq2.R script used in the practical session have to be silenced or you will have an error message!!.*

Now you are ready to do you DEA analysis:

1. Have a look to the p-adj histogram obtained (`res$padj`), what does this result mean?
2. How many genes showed a statistical (p-adj < 0.01) differential expression?. The results has to be justified with a table showing all the altered genes (including, p-val, p-adj, fold change).
3. Taking all this data together, what can you say about the statistical significance of your DEA? Do you feel confident about your differentially expressed genes?

6. Functional prediction

The differential expression results gave you an idea about potential important ***upregulated/overexpressed*** genes. But, what are those genes doing? Using online databases and bioinformatic resources (e.g. PFAM, PHMMER, eggNOG, NCBI Blast, NCBI Taxonomy, STRING-DB, SMART, etc), you need to solve the following questions:

In the computing server, you can find the following files:

```
/home/2019_2020/data/phylo/novel_proteome.faa
(the complete proteome of your target strain. You will need to extract
the protein sequences of your overexpressed genes from here)
```

Tasks:

1. Extract the sequence of each over expressed gene out from its assembled proteome (`novel_proteome.faa`)
2. Save each sequence in FASTA format in an individual file!

Questions:

- Has any other strain of the same abundant species been sequenced? (i.e. whole genome). Report it if so.
- Do all the overexpressed genes have any close homolog in similar strains or in other species/lineages?
- Do the overexpressed genes have any known molecular function (inferred from homologs)? Which function?
- Do the overexpressed genes have any known domain?
- Are the overexpressed genes functionally related? (i.e. protein-protein interactions)
- Could those function you inferred be related with the bloom of algae observed in the hot spring after high-temperature? What would be such relationship? Briefly elaborate your hypothesis.

7. Phylogenetic analysis

To investigate further the functional roles and evolutionary origin of your candidate (upregulated) genes, you decide to perform a phylogenetic analysis comparing them against other prokaryotic proteomes (including the reference genome of the species matching our isolated strain in the hot spring)."

In the computing server, you can find the following files:

```
/home/2019_2020/data/phylo/novel_proteome.faa
(the complete proteome of your target strain. You will need to extract
the protein sequences of your overexpressed genes from here)

/home/2019_2020/data/phylo/all_ref_proteomes.faa
(all reference proteomes concatenate into a single FASTA file,
including the proteome of your target strain. You will need to build a
blast database with it.)

/home/2019_2020/data/phylo/extract_seqs_from_blast_result.py
(a python script that you can use to extract the sequences of a blast
result. Example command line:
$ python extract_seqs_from_blast_result.py  blast_output all_ref_proteomes.faa > homologs.faa
)

/home/2019_2020/data/phylo/additional_seq_info.tsv
(a tab delimited file where columns are:
1: sequence name
2: gene name
3: species name
4: lineage
5: functional description
```

Tasks:
For each over expressed gene (you have individual FASTA files created from previous section), run a standard phylogenetic workflow:

1. Run a blast search for each over expressed protein against all reference proteomes
2. Extract hits with e-value <= 0.00001 (tip: you can use blast parameters for this)
3. Create a FASTA file with all the sequences of selected hits (tip: you can use the extract_sequences_from_blast_result.py)
4. Build a phylogenetic tree out of the fastA file (suggested tools: clustalo, iqtree)
5. Visualize the result (suggested tools: etetoolkit.org/treeview, itol.embl.de, ete3)
6. Using the additional info file located at `additional_seq_info.tsv` (see full path above) extract functional and taxonomic information for each homolog in the trees, and visualize it in the tree for better interpretation.

Questions (for each over expressed gene/protein):

1. What is the closest ortholog from a phylogenetic point of view? From what species?
2. Do orthology assignment support your previous functional annotations? (you might need to look up the functional annotation (i.e. gene names) of close orthologs)
3. Are all genes present in the reference proteome of the same species? Why not? Do they all over expressed genes share the same evolutionary history?

Taking all the project together, **what's your best hypothesis for the effect observed in the hot spring?**