RNA-seq Analysis

Welcome to the World of RNA-seq. In this practical we will explore how to measure gene expression using sequencing data and bioinformatics. This is a critical skill in functional biology. However, most functional biologists rely on collaborators or technicians to analyse their expression data, losing control on how the data is being treated. Fortunately, you are about to learn how to deal with RNA-seq data yourself. You're about to free yourself from the tyranny of evil bioinformaticians!

Throughout this practical, you will have to type some commands in the prompt of our computing cluster. These commands are written in black boxes after the symbol '$', that is, you don't have to type the dollar sign. Whatever is written after without the dollar symbol is the expected output message.

The biology bit

In the previous part you detected which areas of the genome are targeted by p53. The question is: is this having an effect on gene regulation. Well, let's measure gene expression and see if p53 has an impact on it. We have six mouse cells samples. Three of them were treated with doxorubicin, a compound that activates p53, triggering an apoptotic signal. The other three are control (untreated) cells. By comparing the expression profile of these two sample groups, using a technique called differential gene expression, we will find out which genes are activated (or repressed) by p53 at the transcriptional level.

Mapping reads to a reference genome

You will start mapping sequence reads from a paired-end sequencing reaction against the mouse genome (assembly mm9[1]). First we need to have a look at the *fastq* files provided. Using your account at our computing cluster (genome.essex.ac.uk), make sure you are in your home directory directory. The quickest way of doing this is to type this:

```
$ cd ~
```

There is a folder called RNAseq. You can have a look at the files within by typoing this:

```
$ ls -lh RNAseq
total 308M
-rw-r--r-- 1 usr r_amarco  209 Jun  2 11:15 Count_reads_HTSeq.sh
-rw-r--r-- 1 usr r_amarco  252 Jun  2 11:15 Map_reads_TopHat.sh
-rw-r--r-- 1 usr r_amarco 148M Jun  2 11:03 mm9_Ensembl.gtf
-rw-r--r-- 1 usr r_amarco  81M Jun  2 10:56 RNAseq_1.fastq
-rw-r--r-- 1 usr r_amarco  81M Jun  2 10:55 RNAseq_2.fastq
```

1 Each genome assembly has a name. *mm9* is the 9th assembly of the *Mus musculus* (house mouse) genome. Is still used as it is very well annotate, although the most up-to-date version version is *mm10*. Likewise, the most recent human genome assembly is called *hg38*.

You can see two *fastq* files, one ending in '_1.fastq' and another ending in '_2fastq'. These two files correspond to the same paired-ends sequencing run. Each sequence read in the first file has a corresponding pair in the second file, and we need to map the two files together. Have a look at the first *fastq* file using `less`:

```
$ less RNAseq/RNAseq_1.fastq
@SRR1186256.28486137 D25VNACXX130416:1:2315:15140:26810 length=101
TGGGCTCAGCAGCCAATGCCTGCTCACACTCATCCATCTCTTCCTCAGGAGCTGGGGCAGC...
+SRR1186256.28486137 D25VNACXX130416:1:2315:15140:26810 length=101
CC@FFFDFHDGFFIIJJIIIIJIJJIJJJJIIGIGGHGJJJJJJJIJJJJJJJJJ?HIHII...
...
```

Does this format look familiar to you? This is the FASTQ format as described in the slides. Time to map these reads into the reference genome. (Remember, you can exit `less` by pressing 'q'.)

In the previous part you mapped sequence reads from a ChIP-seq reaction with `Bowtie`. As we discussed already, `Bowtie` can't map reads across introns. Fortunately, the program `TopHat` does precisely that. To make `TopHat` map your reads you need a command like this:

```
tophat -o OUTPUT_FOLDER GENOME INPUT_FASTQ_1 INPUT_FASTQ_2
```

The `OUTPUT_FOLDER` is the path of the folder that will contain the output files. In our case we will store the output in ~/mapped_reads ('~' indicates that the output folder will be in your home folder). `GENOME` is the mouse mm9 assembly. The two input files are the two files listed in the RNAseq folder. If you had to run this from your own computer you will write then:

```
tophat -o mapped_reads mm9 RNAseq_1.fastq RNAseq_1.fastq
```

But in real life, datasets are pretty big to be run on a laptop, so we need to use the full power of a computing cluster to do so in a reasonable amount of time (i.e. before we age and die). As in the previous practical, you have this command wrapped in a BASH script. By calling running the script via `qsub`, you will instruct the computing cluster to run `TopHat` for you. So, it's as easy as typing this:

```
$ qsub RNAseq/Map_reads_TopHat.sh
Your job 30720 ("Map_reads_TopHat.sh") has been submitted
```

To make sure that the job is running, check with:

```
$ qstat -u your_username

job-ID  prior   name       user         state submit/start at ...
-----------------------------------------------------------------
  30781 0.00000 Map_reads_ usr          r     06/05/2017 10:25:45
```

Have a look at the script you just submitted:

```
$ cat RNAseq/Map_reads_TopHat.sh
#!/bin/bash
#$ -cwd
#$ -q all.q
#$ -S /bin/bash

# mapping paired-end reads with TohHat to the mouse mm9 genome:
tophat -o ~/mapped_reads mm9 ~/RNAseq/RNAseq_1.fastq
          ~/RNAseq/RNAseq_2.fastq

exit 0
```

What the additional lines of code mean are not part of this shor course, but if you're curious about tit, do not hesitate to ask any of the instructors. The mapping will take approximately 25 minutes so, let's do something else in the meanwhile.

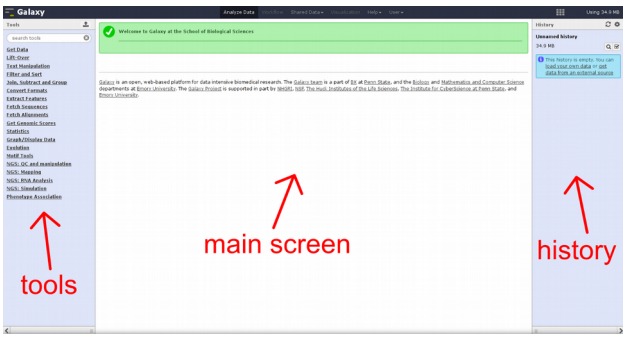
Evaluating the quality of a reads file

For this part we will be using Galaxy. No, it's not an Android phone, it's a web-based tool that runs many bioinformatics' programs in a user-friendly interface. Indeed, the mapping step could have been done in Galaxy as well, but for big datasets it's impractical, and that's why you learned how to run it in a computing cluster. But quality control can be done in Galaxy.

First, open a web browser (like Firefox or Explorer) and to our Galaxy server at:

http://galaxy.essex.ac.uk/

You will see a screen like that:

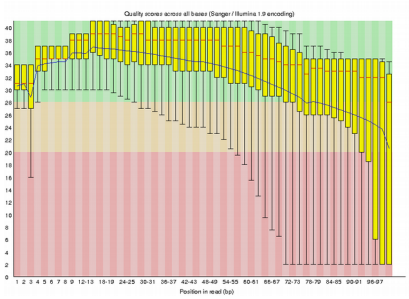


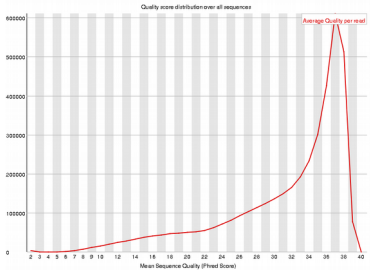
Galaxy is quite intuitive so it doesn't need much introduction. But again, give a shout if need help or do not understand something in particular. First of all, you will need to register a new account.

In the top menu click on 'Shared Data' and then on 'Data libraries'. You will see a folder called 'RNAseq – Proficio'. Go inside. You will see a few files. Select 'RNAseq_1.fastq' by clicking on the open box on the left. Then, click on the option (top part of the page) that says 'to History'. On the pop-up message click on 'Import'. Now, go back to the main page by clicking on the Galaxy logo (top-left). Your history (right window) now shows a fastq file, which is one of the two files you mapped with `TopHat` in the previous step. Have a look at the file by clicking on the 'eye' icon.

Now, in the ‘search tools’ dialog type ‘fastqc’. Click on the program `FastQC`. This is a very useful program used to check the quality of sequencing reads. In the option ‘Short read data from your current history’ you should select the input fastq file (which will be automatically selected in your case). Just click on ‘Execute’, and the server will run `FastQC` for you. Two new files will appear in yout history window, and they will be in yellow while the process is running. In about 5 minutes the files will turn green. They’re ready!

Click on the ‘eye’ icon of the file `FastQC` on data 1: Webpage’. You will see a number of diagnosis plots, the most informative being the ‘Per base sequence quality’ and ‘Per sequence quality scores’. Have a look at them and discuss with colleagues. Your instructor will guide you through them.





How is the quality of the reads overall?

Count reads for each gene

If everything went well, your reads are now mapped to the genome. That’s good news. The bad news is that the mapped reads are in a very tricky format called BAM, which is a binary compressed version of another format called SAM. But you don’t really need to understand these formats as we can convert them to something more intuitive: read counts. That is, we are going to read the output file of `TopHat` and convert it in a table that list all genes in the mouse genome and the number of reads mapped to this gene[2].

`TopHat` produces a few output files. These are in the output folder we defined in the qsub script, in your home folder. You can list them by typing:

```
$ ls -lh ~/mapped_reads
total 56M
-rw-r--r-- 1 amarco r_amarco  27M Jun  2 14:09 accepted_hits.bam
-rw-r--r-- 1 amarco r_amarco  563 Jun  2 14:09 align_summary.txt
-rw-r--r-- 1 amarco r_amarco 120K Jun  2 14:09 deletions.bed
-rw-r--r-- 1 amarco r_amarco 141K Jun  2 14:09 insertions.bed
-rw-r--r-- 1 amarco r_amarco 1.2M Jun  2 14:09 junctions.bed
drwxr-sr-x 2 amarco r_amarco 4.0K Jun  2 14:09 logs
-rw-r--r-- 1 amarco r_amarco  180 Jun  2 13:52 prep_reads.info
-rw-r--r-- 1 amarco r_amarco  28M Jun  2 14:09 unmapped.bam
```

2 The number of reads mapped to a gene is a reflection of the amound of transcript produced, that is, the gene expression level.

For our purposes, most of these files are irrelevant. The reads mapped to the genome are in the file accepted_hits.bam. To find out how many reads are associated to each gene we need this file, but also a file of genome coordinates that tells us where in the genome is each gene. This file is a GTF[3] file. You have a mouse genome annotation GTF file in the RNAseq folder. You can have a look at it with:

```
$ head RNAseq/mm9_Ensembl.gtf
```

To get the read count we can (and we will) use the program `HTSeq`. Again, if you were to run this in your laptop, you will use a command like this:

```
htseq-count -f bam -t CDS ~/mapped_reads/accepted_hits.bam
              RNAseq/mm9_Ensembl.gtf
```

where the `-f bam` option indicates that the input is in BAM format, the `-t CDS` means that we are looking are the coding regions of the genes, and the other two options are the input files. As in the previous part, to run this in a realising time, we will make use of our almighty computing cluster by running a customized script:

```
$ qsub RNAseq/Count_reads_HTSeq.sh
Your job 30735 ("Count_reads_HTSeq.sh") has been submitted
```

That will take a few minutes. At the end, the output file will be in your home directory and will be named something like `Count_reads_HTSeq.oXXXXX` where the X's are some digits[4]. You can start by changing its name to something more meaningful:

```
$ mv Count_reads_HTSeq.sh.oXXXXX read_counts.tab
```

Remember to replace the X's by the actual number in your output file. If you have a quick look at the file you will see that it looks exactly as we expected, a list of genes (or Gene IDs) and the read counts associate to them:

```
$ head read_counts.tab
ENSMUST00000000001      2
ENSMUST00000000003      0
ENSMUST00000000010      0
ENSMUST00000000028      0
ENSMUST00000000033      0
ENSMUST00000000049      0
ENSMUST00000000058      0
ENSMUST00000000080      2
ENSMUST00000000087      0
ENSMUST00000000090      3
```

Why the counts are very small (or 0) for most of the genes? Remember that we are using the cluster because the process would be impractical on a standard PC? Well, actually, the actual process takes

3 General Transfer Format

4 Actually, the digit is the system ID that the computing cluster assigns to your process

days in the cluster, so you've been playing with a small subset of an actual RNAseq file. But that's not a problem as the protocol is exactly the same for bigger files. In the next section we will be using the mapping reads from the actual RNAseq experiments.

Differential Gene Expression

Back to Galaxy. The easiest way to perform a differential gene expression analysis is with the `DESeq2` program installed in our Galaxy server. First, you need to move your files form the cluster to Galaxy. There are several ways of doing this. From your home folder send the *read_counts.tab* file to your University of Essex computer account:

```
$ scp mapped_reads.tab your_username@unix4:pc/desktop
```

and type your University's password. Remember to replace 'your_username' by your real University username. Have a look at the Desktop of your computer and, voilà, the file is there.

Now, go into Galaxy again and click on 'Get Data' and then on 'Upload file'. Now you can either drop the file from the desktop, or 'Choose local file' and find it. Then click on 'Start' and in a few seconds you'll have the file in your Galaxy account.

You may be wondering… where are the other files I need? As I mentioned already, the file you just uploaded was for training purposes, so you know how to map, count reads, and upload to Galaxy yourself. The reality is that if you were to map the actual files it will take you about a week so, we need to speed up the process. Good news is that I mapped these reads for you.

Go again to 'Shared Data'→ 'Data libraries' → 'RNAseq – Proficio'. Select all six files, three 'treated' and three 'untreated', and click on 'to History' → Import. Go back to the main Galaxy page.

Find the `DESeq2` program in the tools window. In the 'Factor' field write 'p53' (or any other meaningful title you may think of). On the box '1:Factor Level' type 'treated' and then select, using the CTRL key, the three 'treated' datasets. On the box '2:Factor Level' type 'untreated' and then select the three 'untreated' datasets. Leave the other options by default and click on 'Execute'. Wait a couple of minutes.

One of the outputs, 'DESeq2 plots', shows a number of plots. Unfortunately, we don't have time in a short course to discuss them, but it's worth that you open them and try to understand. Another output is 'DESeq2 result file', which contains the main output we are going to analyse. Click on the 'eye' icon and you'll see something like that:

GeneID	Base mean	log2(FC)	StdErr	Wald-Stats	P-value	P-adj
Ccng1	10253.9565478971	2.1733415985728	0.0496837029960621	43.7435510542573	0	0
Plau	2868.70628875291	2.29644968758308	0.0591563933877861	38.8199745804182	0	0
Adamts5	2965.14805964652	-3.5324246968983	0.0745015980961711	-47.4140795253604	0	0
Nr4a1	1953.34530631308	3.1725957908854	0.0746751971051276	42.4852683872937	0	0
Ptx3	10991.9420032442	-2.54241242151884	0.0486014725308458	-52.3114278050991	0	0
Icam1	4478.63735905254	2.23008961534929	0.0578959006213177	38.5189554254582	0	0
Notch3	2249.90055725676	2.73860762232716	0.0732767783235185	37.3734719918519	1.05419963864197e-305	1.6483164349909e-302
Epha2	2135.073342786	2.45779451307348	0.0672508448471497	36.5466711780298	2.01401713077518e-292	2.75542718704179e-289
Crip2	1442.08969261518	2.94651026539472	0.0842783818167692	34.9616378705605	8.61827045902016e-268	1.04807744637751e-264
Il6st	12913.6159391834	-1.45668667038069	0.042416124701033	-34.3427571624717	1.80658921536963e-258	1.97731189622206e-255
Mt2	2187.97154447509	-1.97523511200763	0.0620548411881086	-31.8304756597482	2.45248102528859e-222	2.44021862016215e-219
Mki67	8680.16437898843	-1.79983281997386	0.0568558109468329	-31.6560926667095	6.25200113502026e-220	5.70234603523307e-217
Ckap2	5255.93442864738	1.7628614755545	0.0558089037520847	31.5874592947663	5.48925058747985e-219	4.62152674461284e-216

The columns of interest in the output file are: GeneID, the gene names; log2(FC), the fold change of one sample with respect to other (I explain this below); and the P-adj, the p-value corrected for False Discovery Rate.

How to interpret the log2(FC)? First look at the symbol. If it's possitive, the gene is higher expressed in the treated samples than in the untreated one. If it's negative is it's the other way around. Now, look at the magnitude. A value of 1 indicates that expression in one sample is double than expression in the other. If it's 2 the expression is 4 times higher in one sample, if its 3, the expression is 8 times higher, and so forth. For instance, a log2(FC) of -2 indicated that the specific genes is expressed 4 times less in the treated sample than in the untreated one.

Making sense of the results

From here, one can go in many directions, depending on the specific interest. In the last practical you will integrate RNA-seq results with ChIP-seq results. But for completeness, let's do an additional analysis that you may find useful in the future.

Let's find which genes are overexpressed in the 'treated' samples, that is, genes that are activated by p53. In Galaxy find the tool 'Filter' and click on it. Select the 'DESEq2 results file'. We are going to be very strict and select genes with a log2(FC) over 5, and a adjusted p-value of 0.0001 or below. Thus, in the 'With following condition' box we type:

```
c3>5 and c7<=0.0001
```

and we click on 'Execute'.

The output shows the 7 genes that are heavily influenced by p53 activity. Next step (which is not ot be covered here) will be to go to the lab, and perform a few RT-qPCRs to validate these results.

Recommended readings

It's virtually impossible to cover RNA-seq in just a couple of hours (I run a whole semester on that), but I hope you get a fair overview. If you want to learn more about it, there are a few references you may consider. A general textbook is that one:

Pevsner (2015) *Bioinformatics and Functional Genomics*

A more specialised book (yet easy to read) is this:

Korpelainen et al. (2015) *RNA-seq Data Analysis: A practical approach*

But for almost anything you can do in Bioinformatics, ask Google first.