

Bacterial RNAseq Data Analysis Using the MSU HPCC

Benjamin K. Johnson and Robert B. Abramovitch

*Department of Microbiology and Molecular Genetics
Michigan State University
East Lansing, MI, 48824*

Version7_12_14

Table of Contents

Bacterial RNAseq Data Analysis Made Easier Using the HPCC.....	1
Getting an account for the HPCC through iCER.....	2
Downloading the data.....	3
QC and trimming reads – Trimmomatic.....	4
Accessing the HPCC from your computer.....	6
Transferring files to the HPCC from your computer.....	8
Mapping the reads to the genome – Bowtie.....	9
Monitoring the progress of a job with powertools.....	11
Counting the mapped reads – HTSeq.....	12
Differential gene expression with DESeq.....	15
References.....	19

A published example using these methods can be found in Baker *et al.*, 2014 (PMID: 24975990).

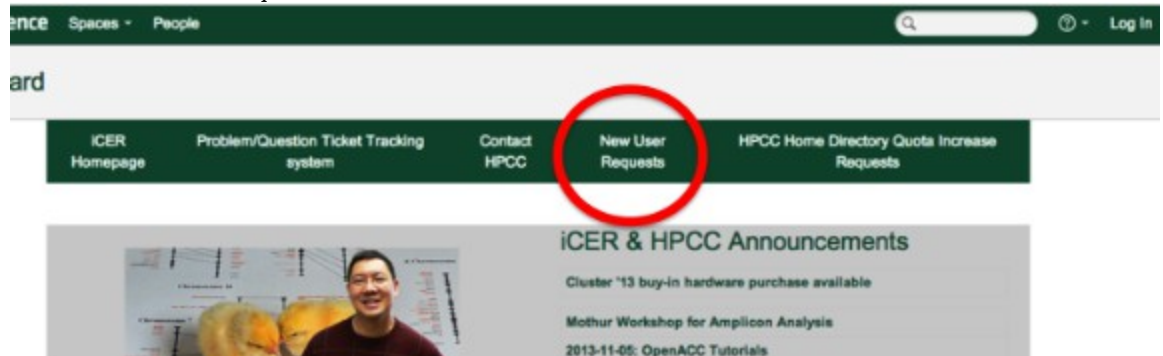
This content is licensed under a creative commons Attribution 4.0 International License
<http://creativecommons.org/licenses/by/4.0/>

Disclaimer: To the best of our knowledge, these methods are correct and complete. However, **we are not responsible for any issues** you may encounter using these methods. It is the end-user's responsibility to ensure the validity of their computational methods and data analyses.

Getting an account for the HPCC through iCER

You will first need an account with iCER to access the HPCC (*High Performance Computing Center*). To do that you will need to either be a faculty member or have one fill out the submission form for you.

1. Go to the HPCC wiki page found [here](#).
 - a. wiki.hpcc.msu.edu
2. Click on the “New User Requests” tab.



3. From there you will be asked to enter your MSU NetID and password.
4. If you are *not* a faculty member, you will see a message telling you that you need to have one make the request for you.
5. If you *are* a faculty member, you will be asked to fill in a series of blanks with who will be responsible for the account and a brief rationale of why that individual needs access (this is for publication purposes when iCER goes to exhibit what cool research is being done on the NSF funded computers).
6. The turn around time is typically less than a day to get an account
 - a. You should receive an e-mail when your account is active

Downloading the data

While you are waiting for your account to be activated, you can download the data from the RNAseq run.

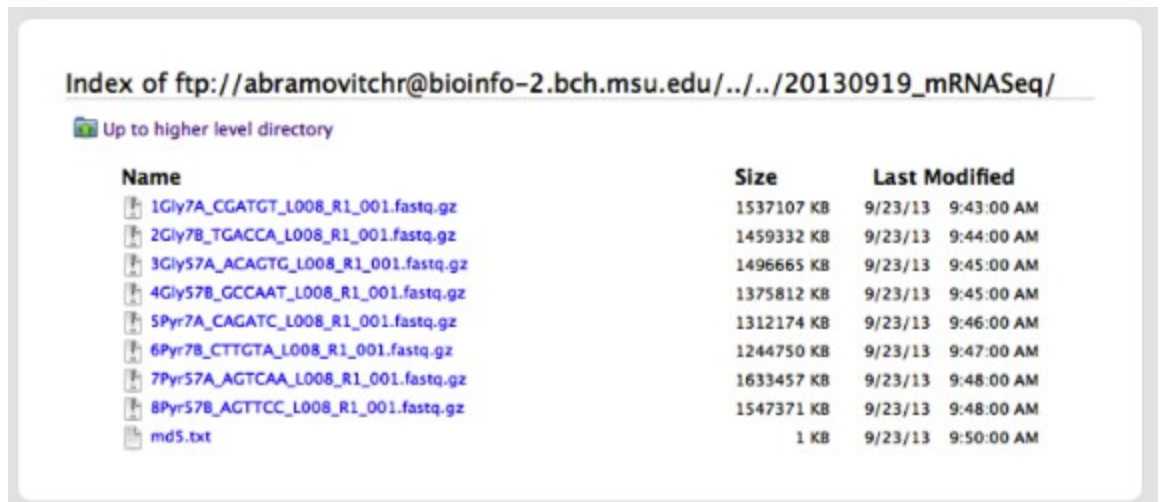
1. You should have received an e-mail from Kevin Carr at the RTSF stating that your sequencing has been completed and you now have access to it through an FTP site. He may have also sent you a link to it. It should look, in general, something like this:

<ftp://username:password@bioinfo-2.bch.msu.edu>

- a. If it asks for a username and password
 - i. Username: will be the “username” in the above link
 - ii. Password: will be the “password” in the above link
2. After you have clicked on the link, you will be presented with a folder containing the data
 - a. If this folder is not the run you were looking for, click on the “Up to higher level directory” tab (this only appears in Firefox and Safari browsers, Chrome will list all the folders at once)



- b. **KEEP IN MIND THAT THE DATA WILL ONLY BE KEPT ON THIS SERVER FOR 30-60 DAYS**
 - c. Older submissions still on the server may be up several, higher directories
3. Once you have found your data folder, you can click on it to view the individual sample files
 - a. They will be compressed to ~1-2 Gb each with a .fastq.gz file suffix and look something like this:



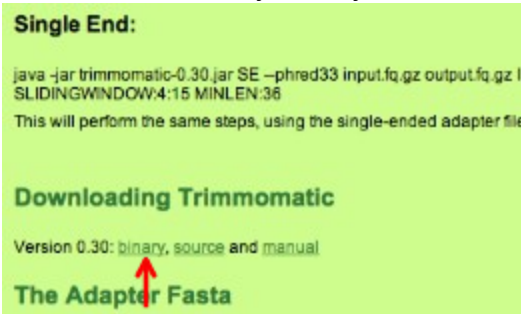
4. Download each sample to a single, new folder (name it something WITHOUT any spaces in it) on your desktop by (in Firefox, Safari, or Chrome) right-clicking and choosing “Save link as” and then choosing the new folder on the desktop.
 - a. These will take a long time to download and it is probably best to have an Ethernet connection to speed things along and ensure you won’t drop the connection.

- b. You do not have to download the “md5.txt” file if you don’t want. It really does not contain any pertinent data for the analysis.

QC and trimming reads – Trimmomatic

Before moving forward to mapping the reads to the genome, it is a ~~good~~ fantastic idea to QC the reads and remove low quality and adapter sequences. We have chosen to do this through a software package called Trimmomatic. This is a commonly used tool for QC/trimming and has been implemented in RNAseq approaches in the literature and is the go to tool at the MSU RTSF.

1. Download the software to your desktop from [here](#).
 - a. <http://www.usadellab.org/cms/?page=trimmomatic>
 - b. Scroll down to where it says “Download Trimmomatic”
 - c. Click on the link that says “binary”



2. This will end up in your Downloads folder and will need to be “unzipped” as it is a compressed file
 - a. This can usually be accomplished simply by double-clicking on the file
3. After this is done (shouldn’t take long), move the folder to your desktop
4. You should now have two folders on your desktop:
 - a. One contains the compressed, raw RNAseq data
 - b. One is the Trimmomatic software
 - i. Should be called “Trimmomatic-0.3x” where the “0.3x” will be the most current version (at the time of writing this it is 0.30)
5. Now it is time to navigate to the command line
 - a. Mac users:
 - i. Go to Applications → then Utilities → and then Terminal (might be worth just dragging it onto the dock)
 - b. Sorry PC users, this tutorial is for Mac/Linux/Unix users. You can do all the trimming and things on the lab (Mac) computer and then transfer the files to your PC for the rest of the steps
6. Trimmomatic runs on Java, so if you haven’t updated to the newest release of [Java](#), that might be a good idea. If you aren’t inclined to download Java, it probably already exists on your computer.
 - a. To check if it already exists, at the command line (Terminal), type `java -showversion`
 - b. If it is on your computer you will get a whole long list of output
 - c. If it doesn’t exist the response will be something along of the lines of “Couldn’t find ‘java -showversion’” or some sort of error.
7. Go to the “Trimmomatic-0.3x” folder on the desktop and click on the subfolder called “adapters”
8. If you have single-end read data:
 - a. Drag and drop the “TruSeq3-SE.fa” out of the “adapters” subfolder and into the main “Trimmomatic-0.3x” folder
 - i. The list of files in the “Trimmomatic-0.3x” folder should now be: adapters (folder), LICENSE, trimmomatic-0.3x.jar, TruSeq3-SE.fa
9. If you have paired-end read data:

- a. Drag and drop the “TruSeq3-PE.fa” out of the “adapters” subfolder and into the main “Trimmomatic-0.3x” folder
 - i. The list of files in the “Trimmomatic-0.3x” folder should now be: adapters (folder), LICENSE, trimmomatic-0.3x.jar, TruSeq3-PE.fa
10. Go back to the command line (Terminal) and we will navigate to the “Trimmomatic-0.3x” folder
11. Type: `cd ~/Desktop/Trimmomatic` and hit the tab key and then hit the enter key
 - a. Tab will autocomplete the rest of the folder name with the version you downloaded
12. You should now be in the “Trimmomatic-0.3x” folder
13. For the sake of the following example, the 0.30 version of Trimmomatic will be used. If you are using a newer version, just substitute that in the place of 0.30 in the following commands
 - a. Also, the following commands are for single-end read data, if you have paired-end read data, check out the Trimmomatic site for the commands for paired-end data [here](#).
 - i. The commands are probably ¾ of the way down the page
14. Type: `java -jar ~/Desktop/Trimmomatic-0.30/trimmomatic-0.30.jar SE -phred33 ~/Desktop/NameOfYourFolderContainingRawRNAseqDataNameOfSample.fastq.gz ~/Desktop/NameOfYourFolderContainingRawRNAseqDataNewFileNameForTrimmedSample.fastq ILLUMINACLIP:TruSeq3-SE.fa:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36`
 - a. The highlighted portions are the only pieces you need to change if you are using Trimmomatic-0.30.
 - b. The “NewFileNameForTrimmedSample.fastq” is an important point to note
 - i. Make the name something simple, descriptive, unique, and starts with “trimmed” and ends with .fastq
 - c. This will initiate the QC/trimming steps and they are as follows:
 - i. Remove adapters
 - ii. Remove leading low quality or N bases (below quality 3)
 - iii. Remove trailing low quality or N bases (below quality 3)
 - iv. Scan the read with a 4-base wide sliding window, cutting when the average quality per base drops below 15
 - v. Drop reads below 36 bases long
 1. (Our reads should be ~50 bases long)
 - d. Feel free to change these values but do so after consulting the Trimmomatic [site](#)
 - e. Repeat this command for each sample, changing the “NameOfSample.fastq.gz” and “NewFileNameForTrimmedSample.fastq” each time
 - f. All the trimmed files will be in the folder that contains all the raw RNAseq data
 - g. **IT IS IMPORTANT TO NOTE THAT YOU MUST MAKE SURE YOU HAVE AT LEAST ENOUGH HARD DRIVE SPACE FOR ~4 Gb PER SAMPLE**
 - i. If you don’t, trim one sample at a time and then transfer the file to the HPCC (see page 7)
 - h. Once the trimming is complete for each sample, some output will appear that will be worth saving in a text file to reference later. It should look something like this:
 - i. TrimmomaticSE: Started with arguments: -phred33 /Volumes/Abramovitch Lab/RNAseq/Jake/1Gly7A_CGATGT_L008_R1_001.fastq.gz /Volumes/Abramovitch Lab/RNAseq/Jake/Trimmomatic/1Gly7Atrimmed.fq.gz ILLUMINACLIP:TruSeq3-SE.fa:2:30:10 LEADING:3 TRAILING:3 SLIDINGWINDOW:4:15 MINLEN:36
Using Long Clipping Sequence:
'AGATCGGAAGAGCGTCGTGTAGGGAAAGAGTGT'
Using Long Clipping Sequence:
'AGATCGGAAGAGCACACGTCTGAACTCCAGTCAC'
ILLUMINACLIP: Using 0 prefix pairs, 2 forward/reverse sequences, 0 forward only sequences, 0 reverse only sequences
Input Reads: 29294470 Surviving: 28401299 (96.95%) Dropped: 893171

(3.05%)
TrimmomaticSE: Completed successfully

Accessing the HPCC from your computer

There are multiple ways with which you can access the HPCC and transfer files to the iCER machines. This overview will be from a GUI standpoint. If you would like to get fancy and learn the Unix commands, you can access the examples on the HPCC wiki [here](#). Any other questions you may have not detailed here (which is a lot...) on how to use the HPCC more effectively, you can check out the user manual [here](#). There are even videos of real-time examples on how to execute different software.

Mac User:

1. At the command line (Terminal), if you type: `ssh YourMSUNetID@hpcc.msu.edu`, you will be prompted for your MSU NetID password. As you begin to type, the cursor will not show that you are entering characters, but you are. Hit the enter key at the end and you will be logged in!
 - a. For me, it would be: `ssh john3434@hpcc.msu.edu`
 - b. Ssh stands for “secure shell”, which means that you are accessing the HPCC computers in a secure, encrypted fashion
 - c. If this is the first time accessing the HPCC, it will send you a warning about not recognizing the RSA fingerprint. Type “yes” or “y” or whatever it needs to agree to continue. It is okay, and necessary, to say you trust iCER to use the HPCC at MSU
 - d. If you are uncomfortable with all of this on your own computer, do all the work on the lab computer
 - e. Once you are logged in your command line should look something like this:



```
benjaminjohnson@john3434@gateway-00:~$ ssh -86x33
Password:
Last login: Wed Sep 25 20:32:58 2013
Load Warning: Did not find: use.cus
Try: "module spider use.cus"

Welcome to Michigan State's High Performance Computing Center
** Unauthorized access is prohibited **

We recommend using dev-ssd09 (or nodes with low usage).
For GPU development please use green nodes.
For HPC development please use underlined nodes.

Development Nodes (usage)          Filesystem Information
-----
dev-intel107 (low)  dev-ssd09 (low)          $(HOME) at 35% usage
dev-intel118 (low)  dev-gfx16 (low)         (used -86G of 250G)
dev-gfx13 (low)     dev-ph112 (low)

Cluster Load (utilization)
-----
short jobs (<= 4 hrs) (97%)      general jobs (82%)
large memory jobs (100%)         gpu jobs (33%)

john3434@gateway-00 ~$
```

- f. Once you’ve logged in, type: `module load powertools`
- g. Then type: `mkdir RNAseq`
 - i. If this is another RNAseq experiment, make a folder with a new name other than RNAseq... Perhaps begin by adding the date: “RNAseq091813”
 - ii. Make sure there are NO spaces in the folder names!
 - iii. Also, if you decide to name the folder OTHER than “RNAseq” make sure you change the line in “alignhtseq.qsub” (see step 13 in HTSeq and change the RNAseq in `cd ~/RNAseq/Bowtie/` to `~/YourFolderName/Bowtie/`)
- h. You’ve just made a new folder called “RNAseq” with the make directory command (`mkdir`)

PC User:

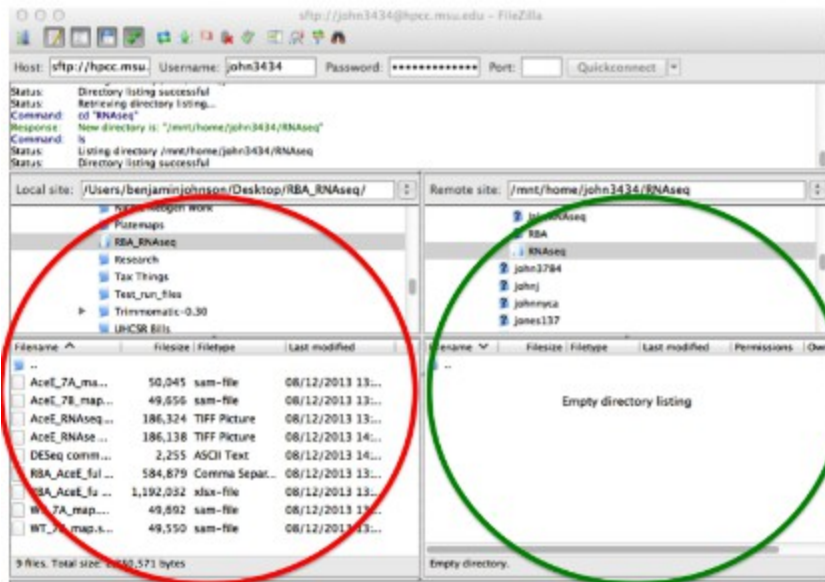
1. I am going to take the easy way out and [here](#) is a video on how to install an ssh client on Windows © (<https://wiki.hpcc.msu.edu/display/hpccdocs/Video+Tutorial+-+Putty>)
 - a. For whatever reason the video doesn’t load, click on the “Download Video” link in the lower left-hand corner of the page.

2. Once you've logged in, type: `module load powertools`
3. Then type: `mkdir RNAseq`
 - a. If this is another RNAseq experiment, make a folder with a new name other than RNAseq... Perhaps begin by adding the date: "RNAseq091813"
 - b. Make sure there are NO spaces in the folder names!
 - c. Also, if you decide to name the folder OTHER than "RNAseq" make sure you change the line in "alignhtseq.qsub" (see step 13 in HTSeq and change the RNAseq in `cd ~/RNAseq/Bowtie/` to `~/YourFolderName/Bowtie/`)
4. You've just made a new folder called "RNAseq" with the make directory command (`mkdir`)

Transferring files to the HPCC from your computer

For this tutorial, we are going to use a GUI client called FileZilla. It really is a fantastic tool that allows files to be uploaded and downloaded from your computer to the HPCC computers.

1. Download FileZilla from [here](https://filezilla-project.org/).
 - a. <https://filezilla-project.org/>
2. Open the application
3. You will need to input a few items at the top of the list to connect to the HPCC
 - a. Host: hpcc.msu.edu
 - b. Username: YourMSU NetID
 - c. Password: YourMSU NetID password
 - d. Port: 22
 - e. Click “Quickconnect”
4. From there, browse to your desktop and the folder containing the trimmed .fastq files (i.e. Users → Yourname → Desktop → Name of the folder with the trimmed files) in the red circled column and browse to the new “RNAseq” folder on the HPCC by double clicking on it in the green circled column



5. Double click on each of the trimmed read files to begin to transfer them to the HPCC under the RNAseq folder.
 - a. Again, this is best if you have an Ethernet connection as it will go substantially faster
6. Once the files are finished transferring, you can then move on to mapping the reads to the genome!
7. **NOTE:** For the following procedures, four additional files are needed (after you have downloaded/renamed the following FASTA and .gtf files you can upload them to the “RNAseq” folder on the HPCC)
 - i. A FASTA(DNA) file with your favorite bacterial genome
 1. For the rest of the processes to work as is, you will need to make sure the file name starts with “trimmed” and doesn’t have any spaces
 - ii. A .gtf file corresponding to your favorite bacterial genome
 1. For the rest of the processes to work as is, you will need to make sure the file name starts with “align” and doesn’t have any spaces
 2. Both of the above files can be downloaded from [here](http://bacteria.ensembl.org/info/website/ftp/index.html). (<http://bacteria.ensembl.org/info/website/ftp/index.html>)

iii. trimmedbowtie.qsub and alignhtseq.qsub

Mapping the reads to the genome – Bowtie

Time for some script editing!

1. Open “trimmedbowtie.qsub” in a text editor (i.e. TextEdit on a Mac)
2. It should look like this:

```
#!/bin/bash -login
#PBS -l walltime=00:20:00,nodes=1:ppn=8,mem=8gb
#PBS -j oe
```

```
cd $PBS_O_WORKDIR
module load bowtie
```

```
mkdir $PBS_JOBID
cp trimmed* ./ $PBS_JOBID
cp align* ./ $PBS_JOBID
cd $PBS_JOBID
```

```
bowtie-build trimmedMtbCDC1551.fa trimmedMtbCDC1551
```

#ppn above should be 1 larger than -p below (thus if ppn=8 up top, the number after '-p' below should be one less)

#the items to change in the script below is the 'trimmedDMSOout_rep2.fastq' section and the subsequent 'alignDMSO-57-2.sam'

#the first file name is the name of the trimmed RNAseq read data that comes from Trimmomatic
#in order for this script to work you need to make sure that whatever the file is named, it starts with 'trimmed' as seen below

#the second file is the file name that will contain the alignment of the reads to the genome. THIS HAS TO BE A .sam FILE!

#every sample = 10 minutes of walltime

#thus, if you have four samples, you will need to change the walltime to 00:40:00 instead of 00:20:00

#this just makes sure that the job will complete

#simply copy and paste each one of these commands below for each sample you want to align with bowtie

#it is as easy as that

```
time bowtie -S -p 7 trimmedMtbCDC1551 trimmedDMSOout_rep2.fastq > alignDMSO-57-2.sam
time bowtie -S -p 7 trimmedMtbCDC1551 trimmedETZout_rep2.fastq > alignETZ-57-2.sam
```

#Do not copy and paste this command for each sample

#Leave as is

```
rm *.fastq
```

3. Please read through the file and the instructions in it
 - a. **NOTE: If you have more than four samples, it is necessary to request more space than the 50 Gb you are given on the HPCC. To do this, navigate back to the HPCC wiki (page 2 of this tutorial) and click on the tab right next to the “New User Requests” tab called “HPCC Home Directory Quota Increase Requests”**
 - b. **To calculate the *minimum* space needed, take the total size of all the unzipped samples and multiply by 3.**
 - i. **As an example, if you had 8 samples, each ~4 Gb in size, the calculation would be 32 Gb x 3 = 96 Gb at MINIMUM needed**
4. The purple highlighted portion is the FASTAfile for your favorite bacterial genome followed by the name given for Bowtie to use

- a. If you aren't doing *M. tuberculosis* CDC1551 RNAseq analysis, then you will need to change the "trimmedMtbCDC1551.fa" to "trimmedYourFavoriteBacterialGenome.fa" (i.e. the FASTA file you downloaded in step 7 on page 8) and to avoid further confusion, you can just leave the rest as is
 - i. If this idea of leaving the rest as is leaves you uneasy, you will need to change the "trimmedMtbCDC1551" to "trimmedYourFavoriteBacterialGenome" and then change the purple highlighted portions in the commands below to "trimmedYourFavoriteBacterialGenome"
- 5. The yellow highlighted portions are the names of your input files (i.e. the trimmed reads files from Trimmomatic). You will need to change these to match the names of your samples. This is the part that is nice if your files start with "trimmed" and are simple/descriptive since you are typing these by hand.
- 6. The green highlighted portions are the names of the output files. Make sure whatever you name these that they start with "align" and end with .sam.
- 7. Copy and paste these commands for each of your samples. When you are done, save and upload this file into the "RNAseq" folder on the HPCC using FileZilla
- 8. Before we run Bowtie, move on to the next section to edit the alignhtseq.qsub file and then come back to step 8.
- 9. Once you've edited and uploaded the alignhtseq.qsub file, type: intel07 and hit the enter key
 - a. This navigates, using the module powertools we loaded earlier, to a cluster named "intel07"
- 10. Then to navigate into the "RNAseq" folder type: cd NameOfFolder (i.e. cd RNAseq)
- 11. Then type: qsub trimmedbowtie.qsub and hit the enter key

Monitoring the progress of a job with powertools

This subset of powertools commands will allow you to monitor the progress of your submitted jobs.

1. Load the module powertools, if you haven't already, by typing: `module load powertools`
2. You now have access to a series of tools that allow you to monitor the progress of a job
 - a. You can list all of the available powertools commands by typing: `powertools`
 - b. The commands you can use to monitor the progress of a job are under the heading "Job Information"
3. To see jobs submitted/actively running/eligible/blocked, you can simply type: `sj`
 - a. This is short for "show job"
 - b. The output will look something like this

```
submitted jobs-----
Job id      Name      User      Time Use S Queue

8 submitted jobs

active jobs-----
JOBID      USERNAME  STATE  PROCS  REMAINING  STARTTIME
15638380   john3434  Running  5      3:25:00    Fri Oct 11 09:43:09
1 active job
5 of 2728 processors in use by local jobs (0.18%)
274 of 387 nodes active (89.25%)

eligible jobs-----
JOBID      USERNAME  STATE  PROCS  VCLIMIT  QUEUE TIME

8 eligible jobs

blocked jobs-----
JOBID      USERNAME  STATE  PROCS  VCLIMIT  QUEUE TIME

8 blocked jobs

Total job: 1

exiting jobs-----
Job id      Name      User      Time Use S Queue
```

- c. The time remaining (i.e. the **walltime** you submitted for the job minus time ran so far) for the job (yellow arrow)
 - d. The job ID (red arrow) can also be used to with other monitoring tools (you can just highlight and copy this number)
4. To receive an e-mail when your job starts, you will need to enter the command "start_monitor" followed by the JOBID
 - a. As an example using the JOBID from above, the command would look like this:
`start_monitor 15638380`
 5. You can also use a command called "qpeek" to look at the output from the running job
 - a. **WARNING: This will generate a LOT of onscreen output**
 - b. As an example using the JOBID from above, the command would look like this: `qpeek 15638380`

Counting the mapped reads – HTSeq

This step in the process will be counting the number of aligned reads per gene and producing a text file that will be read into DESeq to do the differential gene expression analysis.

1. Open “alignhtseq.qsub” in a text editor (i.e. TextEdit on a Mac)
2. It should look something like this:

```
#!/bin/bash -login
#PBS -l walltime=02:20:00,nodes=1:ppn=8,mem=8gb
#PBS -j oe

#Load Numpy module
module load NumPy

#Set the number of threads to match ppn
export MKL_NUM_THREADS=8

#Load HTSeq module
module load HTSeq

#This is changing directories to the folder that you renamed that contains the aligned reads
#In this case I have a folder called Bowtie within the RNAseq folder
#These can be whatever you have chosen to name or rename your folders to be as long as it ends with
the folder that contains the aligned reads from Bowtie
cd ~/RNAseq/Bowtie/

mkdir $PBS_JOBID
cp align* ./$PBS_JOBID
cd $PBS_JOBID

#the items to change in the script below is the 'alignDMSO-57-2.sam' section and the subsequent
'D6DMSOmap2.sam'
#the first file name is the name of the aligned RNAseq reads from the sample 'alignDMSO-57-2' and
the second file is the output that will be a text file that contains the counts of each read that mapped to a
particular gene
#in order for this script to work you need to make sure that whatever the file is named, it starts with
'align' as seen below
#the second file is the file name that will contain the counts of the aligned reads to the genome. THIS
HAS TO BE A .sam FILE!
#every sample = 1 hour and 10 minutes of walltime
#thus, if you have four samples, you will need to change the walltime to 04:40:00 instead of 02:20:00
#this just makes sure that the job will complete
#simply copy and paste each one of these commands below for each sample you want to count with
HTSeq
#it is as easy as that
htseq-count -m intersection-nonempty --stranded=yes alignDMSO-57-2.sam alignMtbCDC1551.gtf >
D6DMSOmap2.sam
htseq-count -m intersection-nonempty --stranded=yes alignETZ-57-2.sam alignMtbCDC1551.gtf >
D6ETZmap2.sam

#Do not copy and paste this command for each sample
#Leave as is
rm *align.sam
3. Take some time to read through the instructions in this file
```

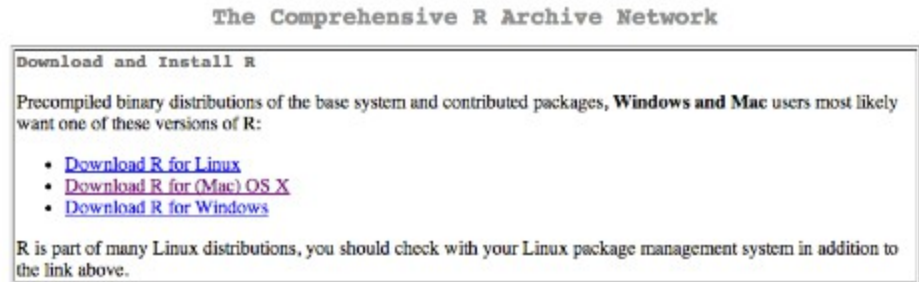
4. The yellow highlighted portion is the output (green highlighted portion on page 8) from Bowtie and you will need to change the name of this file to match the names of output files for your samples
5. The green highlighted portions are the output files from HTSeq. It does not matter what you call these except that they are simple, descriptive, and ends with .sam. Change the names of these to correspond with your samples.
6. The red highlighted portion is stating that your libraries were prepared with a strand specific orientation
 - a. All RNAseq libraries from now (9-27-13) on, unless stated otherwise, that are prepared by the MSU RTSF, are stranded. Thus, the option --stranded=yes is appropriate. If you know your library was prepared without strand specific orientation, change --stranded=yes to --stranded=no
 - b. Also, if you would like to change how a read is counted when aligning to a region/gene in the genome, you can change the blue highlighted portion to something else described [here](#).
7. The purple highlighted portions are the .gtf file for your favorite bacterial genome
 - a. You will need to change the "alignMtbCDC1551.gtf" to "alignYourFavoriteBacterialGenome.gtf" (i.e. the .gtf file you downloaded and renamed in step 7 on page 8)
8. Copy and paste these commands for each of your samples. When you are done, save and upload this file into the "RNAseq" folder on the HPCC using FileZilla
9. This is the point where you can return back to step 8 in the Bowtie analysis
10. **AN ASIDE:**
 - a. To **list** the folders and files within a folder, type: ls
 - b. To **go into** a folder, type: cd NameOfFolder
 - c. To **back out** of a folder, type: cd ..
11. After Bowtie has finished running, it will have produced a folder inside the "RNAseq" folder with a long number, probably something like: 15549846.mgr-04.i. This is the job ID that HPCC gave it. We want to rename this.
12. To rename this folder, navigate into the "RNAseq" folder if you aren't already there and ensure that you are on the intel07 cluster (see step 1f of Accessing the HPCC and step 8 of Bowtie if you aren't already on that cluster)
13. Type: mv HPCCjobIDFolder Bowtie
 - a. Example: mv 15549846.mgr-04.i Bowtie
14. This will rename the folder from the HPCC job ID to Bowtie
15. If you want to name it something else, make sure you change the line (cd ~/RNAseq/Bowtie/) in "alignhtseq.qsub" to whatever you called your folder (i.e. ~/RNAseq/NewFolderName/)
 - a. To do this, you will need to navigate into the new folder by typing: cd ~/RNAseq/NewFolderName/ and hit the enter key
 - b. Then type: nano alignhtseq.qsub
 - i. This will open a text editor in the command line
 - ii. Your mouse won't work here so use the arrow keys to scroll down to the "cd ~/RNAseq/Bowtie/" line
 - iii. Delete the Bowtie name and type in the NewFolderName (remember NO SPACES) so it looks like ~/RNAseq/NewFolderName/
 - iv. Hit control + O and then the enter key
 1. This overwrites the file with the new folder name
 - v. Hit control + X and then the enter key
 1. This will exit from the text editor
16. Now, make sure the alignYourFavoriteBacterialGenome.gtf is in the folder (it should be)
 - a. If not, upload it into the Bowtie (or NewFolderName) subfolder in the "RNAseq" folder using FileZilla
17. Type: qsub alignhtseq.qsub

18. This will create another HPCC job ID folder inside the Bowtie (or NewFolderName) subfolder that will contain the counts per gene per sample (the example from above would have **D6DMSOmap2.sam** and **D6ETZmap2.sam**)
19. Once HTSeq has finished running, use FileZilla to download the count files somewhere on your computer (i.e. the desktop)
 - a. These files will be substantially smaller as they are just text files
20. Now, we can do differential gene expression analysis in DESeq!

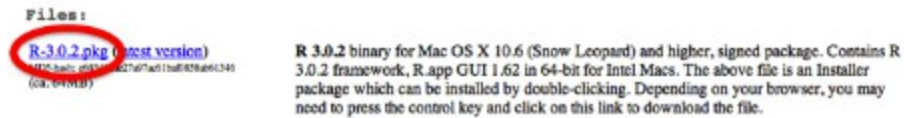
Differential gene expression with DESeq

This tutorial will take you through the basics of doing differential gene expression and the visualization of that data through a software package called DESeq.

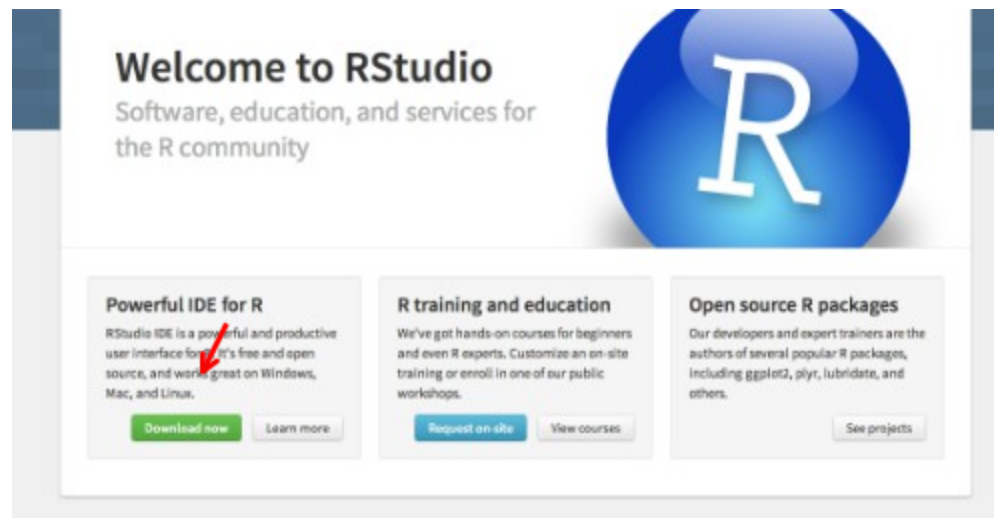
1. First, we need to download two things
 - a. The latest release of R (an open source, statistical software language) [here](#).
 - i. www.r-project.org/
 - ii. Click on the appropriate link for your operating system (Mac OS X, Windows, Linux)




- iii. Then, click on the red circled link, this will begin the download.
- iv. After the download has finished, double-click on the file and follow the instructions to install the software



- b. The latest release of an IDE that works with R [here](#).
 - i. www.rstudio.com/
 - ii. Click on the green button “Download now” under the “Powerful IDE for R”



- iii. Double-click the file after the software has finished downloading and follow the instructions to install the software.
2. Open RStudio
 3. Once everything starts up, type:

- a. `source("http://bioconductor.org/biocLite.R")`
 - b. Then type: `biocLite("DESeq")`
 - c. Then type: `biocLite("DESeq2")`
 - i. These commands will install the packages needed to do analysis
 - d. Then type: `library(DESeq)`
 - e. Then type: `library(DESeq2)`
 - i. The software is now loaded
 - ii. You can accomplish loading new packages by navigating to the “Packages” tab in RStudio and checking the box next to whatever package you want to load
4. Now we can load in the data
- a. It will follow the general pattern:
 - i. `Variablename = read.table("Path to the desktop folder containing the output from HTSeq", row.names=1)`
 - ii. Variablenames should be short and descriptive (you will be typing these again so help yourself out ☺) and contain NO SPACES
 - iii. If you have your folder containing the HTSeq output on the desktop and you have a Unix/Linux based machine like a Mac, the only bit you’ll need to change in the following example is the “benjaminjohnson”, the “RBA_RNAseq” and then the file names (i.e. “AprA_7A_map.sam”)
 1. The “benjaminjohnson” is the user name
 - a. This can be found by opening up your terminal application and looking to see what the name is for your computer in the red underlined portion below
- 
- The screenshot shows a terminal window with the prompt `benjaminjohnson@john3434@gateway-00:~$`. The user name `benjaminjohnson` is underlined in red.
2. The “RBA_RNAseq” is the folder name on the Desktop containing the HTSeq output
 3. The, as an example, “AprA_7A_map.sam” is simply the file name for a particular sample
- b. A specific example for two treatments with two replicates will be shown by typing each of these individually followed by hitting the Enter key (yellow is the variable name and green is the path to the desktop folder on my computer) [The treated samples are aprA1 and aprA2, while the untreated samples are wt1 and wt2]
 - i. `aprA1 = read.table(' /Users/benjaminjohnson/Desktop/RBA_RNAseq/AprA_7A_map.sam', row.names=1)`
 - ii. `aprA2 = read.table(' /Users/benjaminjohnson/Desktop/RBA_RNAseq/AprA_7B_map.sam', row.names=1)`
 - iii. `wt1 = read.table(' /Users/benjaminjohnson/Desktop/RBA_RNAseq/WT_7A_map.sam', row.names=1)`
 - iv. `wt2 = read.table(' /Users/benjaminjohnson/Desktop/RBA_RNAseq/WT_7B_map.sam', row.names=1)`
- c. Now we need to do a minor book keeping step to keep the software not confused (hopefully you aren’t by this step either) by adding names to the list of data by typing each of these individually followed by hitting the Enter key
 - i. `colnames(aprA1) <- 'AprA1'`
 - ii. `colnames(aprA2) <- 'AprA2'`
 - iii. `colnames(wt1) <- 'WT1'`
 - iv. `colnames(wt2) <- 'WT2'`
- d. Now to make our lives easier, we will put all of this data into a single variable named, for this example, ‘rbadat’ by typing this:
 - i. `rbadat <- cbind(wt1, wt2, aprA1, aprA2)`

- e. Now we set up the design matrix for the statistics by typing:
- i. `rbadesign = data.frame(row.names = colnames(rbadat), condition = c('untreated', 'untreated', 'treated', 'treated'), libType = c('single-end', 'single-end', 'single-end', 'single-end'))`
 - ii. This is assigning the design to the variable “rbadesign”. It is defining the first two columns of data (“wt1” and “wt2”) in “rbadat” as the untreated levels and “aprA1” and “aprA2” as the treated levels in the “condition” variable within “rbadesign” (red highlighted portion)
- f. Next we type some more things to help provide the resources for the statistics/inferences:
- i. `singlesamps = rbadesign$libType == 'single-end'`
 - ii. `countTable = rbadat[, singlesamps]`
 - iii. `condition = rbadesign$condition[singlesamps]`
- g. This puts the data into a particular format that DESeq needs to do the analysis:
- i. `cds = newCountDataSet(countTable, condition)`
- h. This is a normalization step:
- i. `cds = estimateSizeFactors(cds)`
 - ii. To see the relative sizes of library size for each sample type: `sizeFactors(cds)`
- i. Estimate dispersions (variances) between samples:
- i. `cds = estimateDispersions(cds)`
- j. Visualize the dispersion estimates:
- i. `DESeq:::plotDispEsts(cds)`
- k. Do the statistics!
- i. `res = nbinomTest(cds, "untreated", "treated")`
- l. To visualize the results, we need to correct a script in DESeq, so type:
- i. `fix(plotMA)`
 - ii. Then you are presented with some lines of code in R (oooo... ahhhh...)
 - iii. Delete all of them and copy and paste the following starting at the word function (if the “1.” comes through the copy and paste, delete it and all the space before “function” (see below for a screenshot):
1.

```
function(x, ylim, col = ifelse(x$padj >= 0.05, "gray32", "red3"),
      linecol = "#ff000080", xlab = "Mean of normalized counts",
      ylab = 'Expression ratio', log = "x", cex = 0.45,
      ...)
{
  if (!(is.data.frame(x) && all(c("baseMean", "log2FoldChange")
%in%
      colnames(x))))
    stop("'x' must be a data frame with columns named 'baseMean',
'log2FoldChange'.")
  x = subset(x, baseMean != 0)
  py = x$foldChange
  if (missing(ylim))
    ylim = c(0, 4) * quantile(abs(py[is.finite(py)]), probs = 0.99) * 1.1
  #ylim = c(0.05, 10)
  plot(x$baseMean, pmax(ylim[1], pmin(ylim[2], py)), log = log,
      pch = ifelse(py < ylim[1], 6, ifelse(py > ylim[2], 2,
      16)), cex = cex, col = col, xlab = xlab, ylab
= ylab,
      ylim = ylim, ...)
  abline(h = 1, lwd = 4, col = linecol)
}
```

```

1 function (x, ylim, col = ifelse(x$pval >= 0.05, "gray32", "red3"),
2       linecol = "#ff0000", xlab = "mean of normalized counts",
3       ylab = 'Expression ratio (relative to DMSO control)', log = "x", cex = 0.45,
4       ...)
5 {
6   if (!(is.data.frame(x) && all(c("baseMean", "log2FoldChange") %in%
7     colnames(x))))
8     stop("'x' must be a data frame with columns named 'baseMean', 'log2FoldChange'.")
9   x = subset(x, baseMean != 0)
10  py = x$log2FoldChange
11  if (missing(ylim))
12    ylim = c(0, 4) + quantile(obs(py[is.finite(py)]), probs = 0.99) * 1.1
13  @ylim = c(0.05, 10)
14  plot(x$baseMean, pmax(ylim[1], pmin(ylim[2], py)), log = log,
15       pch = ifelse(py < ylim[1], 6, ifelse(py > ylim[2], 2,
16         16)), cex = cex, col = col, xlab = xlab, ylab = ylab,
17       ylim = ylim, ...)
18  abline(h = 1, lwd = 4, col = linecol)
19 }
20

```

- m. Now, to actually see what you've done, type:
 - i. `plotMA(res, ylim=c(0.05, 10), log='xy')`
 1. To change the values for the y-axis, you can just alter the (0.05, 10) to whatever values you'd like. This is on a log scale.
 - ii. If that doesn't work for some reason, try:
 1. `DESeq:::plotMA(res, ylim=c(0.05, 10), log='xy')`
 - iii. The red dots indicate a p-value < 0.05 and the black dots are the converse
- n. To export all of this fun information into something readable by Excel, type:
 - i. `write.csv(res,`
`file='/Users/benjaminjohnson/Desktop/RBA_RNAseq/RBA_AprA_fullresults.csv')`
 - ii. Again, you will need to change the path components as you did in Step 4b to navigate to your desktop folder. The "RBA_AprA_fullresults.csv" is simply the name you give the comma-separated file (i.e. .csv). Thus, you can call this whatever you'd like as long as it ends with .csv and doesn't have any spaces.
 - iii. You can open this file in Excel!
- o. If you would like to do analysis in another software package, you can export all of the reads per gene data to a comma-separated file by doing the following:
 - i. Type: `write.csv(countTable, file=`
`'/Users/benjaminjohnson/Desktop/RBA_RNAseq/AprACountsPerGene.csv')`
 - ii. Again, you will need to change the path components as you did in Step 4b to navigate to your desktop folder. The "AprACountsPerGene.csv" is simply the name you give the comma-separated file (i.e. .csv). Thus, you can call this whatever you'd like as long as it ends with .csv and doesn't have any spaces.

References

1. Bolger, A. M., Lohse, M., & Usadel, B. (2014). Trimmomatic: A flexible trimmer for Illumina Sequence Data. *Bioinformatics* btu170.
2. Langmead, B., Trapnell, C., Pop, M., and Salzberg, S.L. (2009). Ultrafast and memory-efficient alignment of short DNA sequences to the human genome. *Genome Biol* 10:R25.
3. Anders, S., Pyl, P. T., and Huber, W. (2014). HTSeq — A Python framework to work with high-throughput sequencing data. *bioRxiv preprint* [doi: 10.1101/002824](https://doi.org/10.1101/002824)
4. Anders, S., and Huber, W. (2010). Differential expression analysis for sequence count data. *Genome Biol* 11:R106.