MCBL Documentation

Release 1.0

Saranga

February 19, 2016

So far,

1. *Overview of Next Generation Sequencing*
2. *Setting an iPlant account*
3. *Uploading data to iPlant*
4. *Intrdouction to iPlant Discovery Environment*
5. *QC and cleaning sequence data*
6. *Mapping of reads to the genome*
7. *Assembling transcripts and estimating their abundances*

Today,

1. *Introduction to iPlant Atmosphere*
2. *Introduction to Terminal*
3. *Doing down stream analysis in the Terminal*

CHAPTER 1

Introduction to iPlant Atmosphere

- iPlant Atmosphere:Introduction Introduction to iPlant Atmosphere, this includes:
- What is iPlant Atmosphere ?
- Requesting Access to Atmosphere
- Logging In to and Signing Out of Atmosphere
- Using Instances
  - Launching a New Instance
  - Logging in to an Instance
  - Rebooting, Stopping and restarting an instance, and Suspending an instance

CHAPTER 2

Introduction to Terminal

2.1 Overview of Linux

Linux is a free OS and very similiar to the UNIX OS in terms of concepts and features.

Linux Distributions

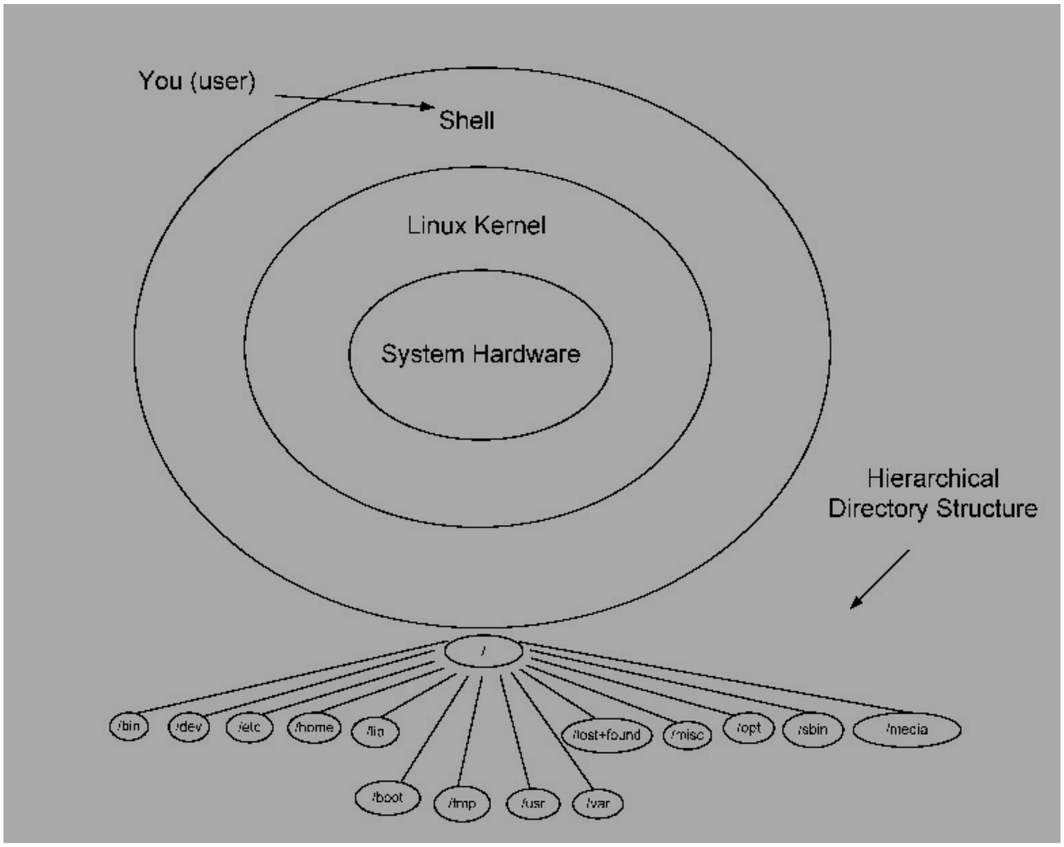
2.1.1 Linux System Structure

Linux system has three main components:

Kernel It controls system hardware including memory, processors, disks, and I/ O (Input/ Output) devices. It schedules processes, enforces security, manages user access, and so on. The kernel receives instructions from the shell, engages appropriate hardware resources, and acts as instructed.

Shell (This the important part for our class) The shell is a program that accepts and interprets text-mode commands. The user provides instructions (commands) to the shell, which are interpreted and passed to the kernel for processing.

Hierarchical directory structure Linux uses the conventional hierarchical directory structure where directories may contain both files and sub-directories. Sub-directories may further hold more files and sub-directories. A subdirectory, also referred to as a child directory, is a directory located under a parent directory. >



- /home/ username/dir1/ subdir1 -root (parent of *home*)
- home - sub-directory or child of / (*root*)

2.2 Starting a Shell

- Through SSH
- Using graphical interface

[] prompt, waiting for you to start entering commands.

2.3 Terminal Commands

2.3.1 *pwd (Print Working Directory)*

When you first login, you are logged into your home directory **(/home/username)**.

To find out what is your current working directory, type

```
$ pwd
/home/kiriya
```

2.3.2 *mkdir (makding a directory)*

To make a subdirectory called *Software* in your home directory, type

```
$ mkdir Software
```

2.3.3 *ls (list)*

To see what is inside the home directory, type

```
$ ls
```

2.3.4 *cd (change directory)*

To change the current directory to the "Software", type

```
$ cd Software
```

:~$ cd ../ -by typying this you can go back to where you started.

2.3.5 Excercise

Use the Terminal commands we already learned to do the following steps.

1. Creat following directory structure in your "Home Directory"

   RNA-Seq/Reference/Genome

   RNA-Seq/Reference/Annotation

   RNA-Seq/RAW_Data

   RNA-Seq/Adapters

   RNA-Seq/QC/Fastqc_Out

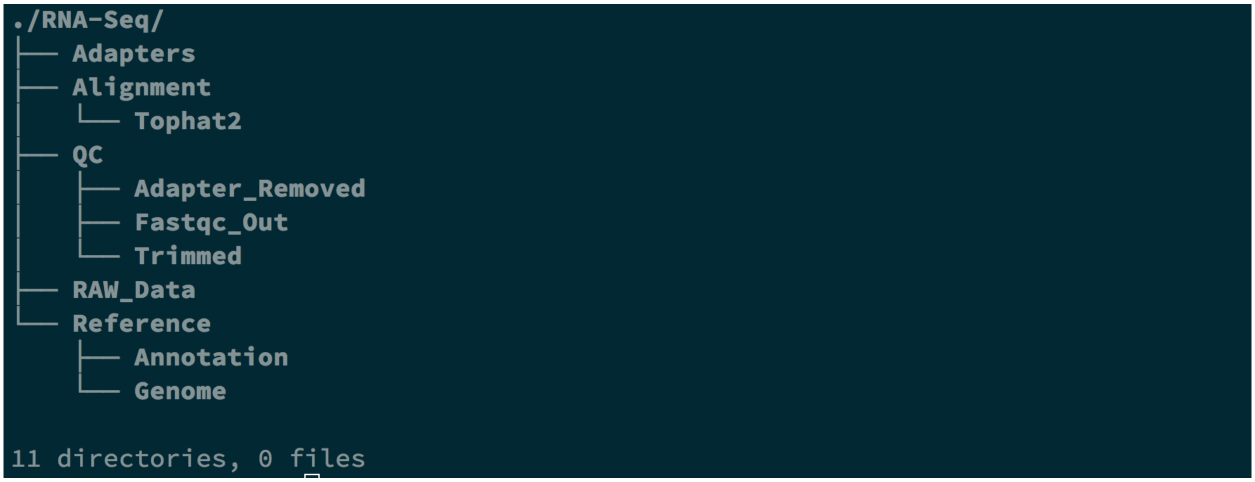
   RNA-Seq/QC/Adapter_Removed

   RNA-Seq/QC/Trimmed

   RNA-Seq/Alignment/Tophat2

Note: You might have to use "-p" option to create non-exsisting intermediate directories**

Final output:



```
./RNA-Seq/
├── Adapters
├── Alignment
│   └── Tophat2
├── QC
│   ├── Adapter_Removed
│   ├── Fastqc_Out
│   └── Trimmed
├── RAW_Data
└── Reference
    ├── Annotation
    └── Genome

11 directories, 0 files
```

2.4 File Handling Through the Terminal

2.4.1 Displaying Content of a Compressed gunzip File

zcat [filename.gz]

```
$ zcat sequence.fastq.gz | less
```

2.4.2 De-compressing gunzip File

gzip -d [filename.gz]

```
$ gzip -d sequence.fastq.gz
```

2.4.3 Displaying Content of a File

cat display whole content of a file on the screen

less display contents of a file onto the screen a page at a time

head display first ten lines of a file to the screen

tail display last ten lines of a file to the screen

cat [filename]

```
$ cat sequence.fastq | less
```

less [filename]

```
$ less sequence.fastq
```

head [filename]

```
$ head sequence.fastq
```

tail [filename]

```
$ tail sequence.fastq
```

2.4.4 Renaming a File

mv [orginalfile.txt] [newnamefile.txt]

```
$ mv sequence.fastq new_sequence.fastq
```

2.4.5 Searching the Contents of a File

grep [options] [word_to_find] [filename]

```
$ grep "@" sequence.fastq
  @D00109:408:C77LEANXX:2:1101:1715:1962 1:N:0:18
```

2.4.6 Concatenating two or more files

cat [fist_file.txt] [second_file.txt] [thrid_file.txt] .... [N_file.txt] > [output_file.txt]

```
$ cat first.txt second.txt > third.txt
```

2.4.7 Finally, Compress that File!!

gzip [filetocompress]

```
$ gzip sequence.fastq
```

2.4.8 Excercise

1. Count the number of sequnces in a fastq.gz file

Note: Use zcat and pip (“|”) the output to **grep -c** [word_to_grep]

CHAPTER 3

How to Install Software on Linux

1. *Install Software From Your Distribution's Repositories*
2. *Downloading and Unpacking a Binary Archive*
3. *Compileing From Source*

3.1 Software we need

3.1.1 *Quality Control*

- Fastqc
- sickle
- scythe

3.1.2 *Alignment Software*

- Tophat2

3.2 Ways to Install Software

3.2.1 1. Install Software From Your Distribution's Repositories

First Search:

sudo apt-cache policy [software_name]

```
$ sudo apt-cache policy fastqc
 [sudo] password for swijeratne:
 fastqc:
   Installed: (none)
   Candidate: 0.10.1+dfsg-2
   Version table:
      0.10.1+dfsg-2 0
         500 http://us.archive.ubuntu.com/ubuntu/ trusty/universe amd64 Packages
```

Warning: Not all the Linux distributions have fastqc in their repos. If you see *Unable to locate package* warning you have to use other methods described in this class to install your software.

Then Install:

```
$ sudo apt-get install fastqc
```

If you see *Unable to locate package* massage, go to *Compileing From Source* and read that section first. Then, install *fastqc*

3.2.2 2. Downloading and Unpacking a Binary Archive

To download tophat2 binaries, from your home directory type

```
$ cd Software
```

Then,

```
$ wget https://ccb.jhu.edu/software/tophat/downloads/tophat-2.1.0.Linux_x86_64.tar.gz
```

```
$ tar -xvf tophat-2.1.0.Linux_x86_64.tar.gz
```

```
$ cd tophat-2.1.0.Linux_x86_64/ && ls -ls
```

To execute tophat2,

```
$ ./tophat2
```

3.2.3 3. Compileing From Source

Go back to *Software* directory by typing,

```
$ cd ../
```

Download sickle and and scythe

```
$ wget https://github.com/najoshi/sickle/archive/master.zip
```

or to download github repo,

```
$ git clone https://github.com/najoshi/sickle.git
```

Unzip master file if you use *wget* method

```
$ unzip master.zip
```

Remove master.zip from your directory

```
$ rm master.zip
```

Note: If you clone the github repo you can skip above steps

Clone *scythe* using "git clone" command

```
$ git clone https://github.com/najoshi/scythe.git
```

Compile sickle and scythe

```
$ cd sickel-master
```

```
$ make
```

```
$ ls -ls
```

Do the same for the scythe,

```
$ cd scythe
```

```
$ make all
```

```
$ ls -ls
```

Now, add both binaries to *PATH*, so you can access them anywhere,

```
$ sudo ln -s /home/yourusername/RNA-Seq/Software/sickel-master/sickle /usr/local/bin
$ sudo ln -s /home/yourusername/RNA-Seq/Software/scythe//scythe /usr/local/bin
```

Install fastqc from source

```
$ wget http://www.bioinformatics.babraham.ac.uk/projects/fastqc/fastqc_v0.11.3.zip

$ unzip fastqc_v0.11.3.zip

$ cd  ~/RNA-Seq/Software/FastQC  (Assuming your files inside RNA-Seq/SoftwareFastQC)

$ chmod a+x ./fastqc (make fastqc executable)

$ sudo ln -s  ~/RNA-Seq/Software/FastQC/fastqc  /usr/local/bin/fastqc (make a link to /usr/local/bin)
```

CHAPTER 4

Data Analysis in the Terminal

4.1 Quality Control

4.1.1 *Quaulity Check With Fastqc*

To get help,

```
$ fastqc --help
```

```
            FastQC - A high throughput sequence QC analysis tool

SYNOPSIS

    fastqc seqfile1 seqfile2 .. seqfileN

    fastqc [-o output dir] [--(no)extract] [-f fastq|bam|sam]
           [-c contaminant file] seqfile1 .. seqfileN

DESCRIPTION

    FastQC reads a set of sequence files and produces from each one a quality
    control report consisting of a number of different modules, each one of
    which will help to identify a different potential type of problem in your
    data.

    If no files to process are specified on the command line then the program
    will start as an interactive graphical application.  If files are provided
    on the command line then the program will run with no user interaction
    required.  In this mode it is suitable for inclusion into a standardised
    analysis pipeline.

    The options for the program as as follows:

    -h --help       Print this help file and exit

    -v --version    Print the version of the program and exit

    -o --outdir     Create all output files in the specified output directory.
                    Please note that this directory must exist as the program
                    will not create it.  If this option is not set then the
                    output file for each sequence file is created in the same
                    directory as the sequence file which was processed.
    --casava        Files come from raw casava output. Files in the same sample
                    group (differing only by the group number) will be analysed
                    as a set rather than individually. Sequences with the filter
                    flag set in the header will be excluded from the analysis.
                    Files must have the same names given to them by casava
                    (including being gzipped and ending with .gz) otherwise they
                    won't be grouped together correctly.

    --nano          Files come from naopore sequences and are in fast5 format. In
                    this mode you can pass in directories to process and the program
                    will take in all fast5 files within those directories and produce
                    a single output file from the sequences found in all files.

    --nofilter      If running with --casava then don't remove read flagged by
                    casava as poor quality when performing the QC analysis.

    --extract       If set then the zipped output file will be uncompressed in
                    the same directory after it has been created.  By default
                    this option will be set if fastqc is run in non-interactive
                    mode.

    -j --java       Provides the full path to the java binary you want to use to
                    launch fastqc. If not supplied then java is assumed to be in
                    your path.

    --noextract     Do not uncompress the output file after creating it.  You
                    should set this option if you do not wish to uncompress
                    the output when running in non-interactive mode.

    --nogroup       Disable grouping of bases for reads >50bp. All reports will
                    show data for every base in the read.  WARNING: Using this
                    option will cause fastqc to crash and burn if you use it on
                    really long reads, and your plots may end up a ridiculous size.
                    You have been warned!

    -f --format     Bypasses the normal sequence file format detection and
                    forces the program to use the specified format.  Valid
                    formats are bam,sam,bam_mapped,sam_mapped and fastq

    -t --threads    Specifies the number of files which can be processed
                    simultaneously.  Each thread will be allocated 250MB of
                    memory so you shouldn't run more threads than your
                    available memory will cope with, and not more than
                    6 threads on a 32 bit machine

    -c              Specifies a non-default file which contains the list of
    --contaminants  contaminants to screen overrepresented sequences against.
                    The file must contain sets of named contaminants in the
                    form name[tab]sequence.  Lines prefixed with a hash will
                    be ignored.

    -a              Specifies a non-default file which contains the list of
    --adapters      adapter sequences which will be explicity searched against
                    the library. The file must contain sets of named adapters
                    in the form name[tab]sequence.  Lines prefixed with a hash
                    will be ignored.

    -l              Specifies a non-default file which contains a set of criteria
    --limits        which will be used to determine the warn/error limits for the
                    various modules.  This file can also be used to selectively
                    remove some modules from the output all together.  The format
                    needs to mirror the default limits.txt file found in the
                    Configuration folder.

   -k --kmers       Specifies the length of Kmer to look for in the Kmer content
                    module. Specified Kmer length must be between 2 and 10. Default
                    length is 7 if not specified.

   -q --quiet       Supress all progress messages on stdout and only report errors.

   -d --dir         Selects a directory to be used for temporary files written when
                    generating report images. Defaults to system temp directory if
                    not specified.

BUGS

    Any bugs in fastqc should be reported either to simon.andrews@babraham.ac.uk
    or in www.bioinformatics.babraham.ac.uk/bugzilla/
```

```
$ cd RNA-Seq/QC/Fastqc_Out
```

Code For Few Samples

```
$ fastqc -t 4  --outdir  ~/RNA-Seq/QC/Fastqc_Out RNA-Seq/RAW_Data/3290-TM-0001-18_S18_L002_R1_001-2.f
```

Code For Many Samples

Note: If your raw data path names end with *.fastq change the *.fastq.gz to *.fastq* in the following code.

```
$ for f in ~/RNA-Seq/RAW_Data/*.fastq.gz; do fastqc --outdir  ~/RNA-Seq/QC/Fastqc_Out -t 4 $f  ; done
```

Explanation

```
$ for f in ~/RNA-Seq/RAW_Data/*.fastq.gz;
```

Note: This will pick any file that has file extension .fastq.gz in the */home/yourusername/RNA-Seq/RAW_Data* directory.

Then,

```
$ do fastqc --outdir  ~/RNA-Seq/QC/Fastqc_Out -t 4 $f
```

Note: will execute fastqc on each file in the /home/yourusername/RNA-Seq/RAW_Data until there is no more .fastq.gz files left in that directory.

```
$ cd ~/ #Go back to home directory
```

4.1.2 *Adapter Trimming with scythe*

```
$ scythe --help
```

```
Usage: scythe -a adapter_file.fasta sequence_file.fastq
Trim 3'-end adapter contaminants off sequence files. If no output file
is specified, scythe will use stdout.

Options:
  -p, --prior               prior (default: 0.300)
  -q, --quality-type        quality type, either illumina, solexa, or sanger (default: sanger)
  -m, --matches-file        matches file (default: no output)
  -o, --output-file output trimmed sequences file (default: stdout)
  -t, --tag         add a tag to the header indicating Scythe cut a sequence (default: off)
  -n, --min-match   smallest contaminant to consider (default: 5)
  -M, --min-keep    filter sequnces less than or equal to this length (default: 35)
  --quiet           don't output statistics about trimming to stdout (default: off)
  --help            display this help and exit
  --version         output version information and exit

  Information on quality schemes:
  phred                     PHRED quality scores (e.g. from Roche 454). ASCII with no offset, range:
  sanger            Sanger are PHRED ASCII qualities with an offset of 33, range: [0, 93]. From
                    NCBI SRA, or Illumina pipeline 1.8+.
  solexa            Solexa (also very early Illumina - pipeline < 1.3). ASCII offset of
                    64, range: [-5, 62]. Uses a different quality-to-probabilities conversion than ot
                    schemes.
  illumina          Illumina output from pipeline versions between 1.3 and 1.7. ASCII offset of 64,
                    range: [0, 62]
```

Unzip your data before this step,

gzip -d Code For few Samples

```
$ gzip -d RNA-Seq/RAW_Data/3290-TM-0001-18_S18_L002_R1_001-2.fastq.gz
$ gzip -d RNA-Seq/RAW_Data/3290-TM-0001-18_S18_L004_R1_001-2.fastq.gz
```

Note: Your outputs will be under RNA-Seq/RAW_Data/

gzip -d Code For Many Samples

Note: You have to be in your *HOME* directory to issue following commands. If are not in your *HOME* do,

```
$ cd ~/
```

to go back to your *HOME*.

```
$ for f in RNA-Seq/RAW_Data/*.gz; do gzip -d  $f  ; done
```

Scythe Code For Few Samples

```
$ scythe  -a RNA-Seq/Adaptors/TruSeq_adapters.fasta  -M 50 -o RNA-Seq/QC/Adapter_Removed/Adapt_rem_3

$ scythe  -a RNA-Seq/Adaptors/TruSeq_adapters.fasta  -M 50 -o RNA-Seq/QC/Adapter_Removed/Adapt_rem_3
```

Scythe Code For Many Samples

```
$ for f in RNA-Seq/RAW_Data/*.fastq; do scythe -a RNA-Seq/Adaptors/TruSeq_adapters.fasta -o RNA-Seq/
```

4.1.3 *Quality Trimming with sickle*

```
sickle se --help
```

```
Usage: sickle se [options] -f <fastq sequence file> -t <quality type> -o <trimmed fastq file>

Options:
-f, --fastq-file, Input fastq file (required)
-t, --qual-type, Type of quality values (solexa (CASAVA < 1.3), illumina (CASAVA 1.3 to 1.7), sanger
-o, --output-file, Output trimmed fastq file (required)
-q, --qual-threshold, Threshold for trimming based on average quality in a window. Default 20.
-l, --length-threshold, Threshold to keep a read based on length after trimming. Default 20.
-x, --no-fiveprime, Don't do five prime trimming.
-n, --trunc-n, Truncate sequences at position of first N.
-g, --gzip-output, Output gzipped files.
--quiet, Don't print out any trimming information
--help, display this help and exit
--version, output version information and exit
```

```
$ sickle se -q 20  -t sanger -f RNA-Seq/QC/Adapter_Removed/Adapt_rem_3290-TM-0001-18_S18_L002_R1_001
```

Sickle Code For Many Samples

```
$ for f in RNA-Seq/QC/Adapter_Removed/*.fastq; do sickle se -q 20  -t sanger  -f $f -o RNA-Seq/QC/Tr
```

4.2 Short-reads Alignment with Tophat2

4.2.1 Indexing your Genome

To make bowtie2 indexes for your Genome,

```
$ cd RNA-Seq/Reference/Genome/
```

```
$ gzip -d Gmax_275_v2.0.gz
```

```
$ mv Gmax_275_v2.0 Gmax_275_v2.0.fa
```

```
$ bowtie2-build Gmax_275_v2.0.fa Gmax_275_v2.0
```

Warning: THIS WILL TAKE LONG TIME

4.2.2 Aligning Short Reads

To align short reads to Genome using Tophat2,

```
$ cd ~/RNA-Seq
```

```
$tophat2 --num-threads 4  --output-dir RNA-Seq/Alignment/Tophat2 RNA-Seq/Reference/Genome/Gmax_275_v2
```

Tophat2 Code For Many Samples

```
$for f in RNA-Seq/QC/Trimmed/*.fastq; do  tophat2 --num-threads 4  --output-dir RNA-Seq/Alignment/${
```

4.3 Excercise

1. Run Cufflinks2 on alignment file(SAM)

CHAPTER 5

Indices and tables

- genindex
- modindex
- search

Python Module Index

Index