Raw Read QA/QC

Get the data

Download the reads that we will be using for this workshop found in `/data/genomics/workshops/smsc/RNA_Seq/` . Make a new subdirectory in your `/pool/genomics` directory and copy the reads there.

```
$ cd /pool/genomics/<username>
$ mkdir RNAseq_SMSC
$ cd RNAseq_SMSC
$ cp /data/genomics/workshops/smsc/RNA_Seq/SMSC_data.tar.gz .
```

Now unpack/unzip the tar file:

```
$ tar -xvzf SMSC_data.tar.gz
```

Go ahead and take a look at the data in your directory:

```
ls -lh data
```

Most of the data used in this tutorial is data generated as part of the Red Siskin Genome Project. Note that for each replicate (Brain, Embryo, Eye & Femur), there are two files, ending in: _1.fastq and _2.fastq. This is because the reads are paired end.

In this tutorial we are assuming that there is no good reference genome (even though one exists). Because of this, we need to generate a reference transcriptome that includes all of the data that you wish to analyze, so our *de novo* transcriptome assembly will include the reads from all of the replicates.

However, before we start the Trinity run, we will do some quality assessment with FASTQC.

Read quality assessment with FASTQC

FastQC is a program that can quickly scan your raw data to help figure out if there are adapters or low quality reads present. Create a job file to run FastQC on one of the eight raw read files you downloaded (eg. data/RNA_Brain_1.fastq.gz).

- Create a job file to run FASTQC on the data you just copied to your working directory:

- Use the QSub Generator
- *Remember Chrome works best, but you'll need to accept the security warning message*
- **CPU time:** short *(we will be using short for all job files in this tutorial)*
- **memory:** 2GB
- **PE:** serial
- **module:** `bioinformatics/fastqc/0.11.5`
- **command:** `fastqc <FILE.fastq>`
- **job name:** FASTQC *(or name of your choice)*
- **log file name:** FASTQC.log
- hint: either use `nano` or upload your job file using `scp` from your local machine into the `assembly_tutorial` directory. See here and here on the Hydra wiki for more info.
- hint: submit the job on Hydra using `qsub`
- after your job finishes, find the results and download some of the images, e.g. `per_base_quality.png` to your local machine using `scp`

Trimming adapters with TrimGalore

TrimGalore will auto-detect what adapters are present and remove very low quality reads (quality score <20) by default.

Note: We will not be doing this step for this particular workshop because the data has already been trimmed. However, remember that trimming adapters should always be done prior to genome/transcriptome assembly!

- Create a job file to run TrimGalore on your data:

  - **command**: `trim_galore --paired --retain_unpaired <FILE_1.fastq> <FILE_2.fastq>`
  - **module**: `bioinformatics/trimgalore/0.4.0`
  - You can then run FastQC again to see if anything has changed.