RNA-seq quantification from published data

Thomas Dugé,Franziska Liesecke

Abstract

This workflow described how to download RNA-seq raw data from NCBI or EBI and to process them to quantify transcript abundance.

Citation: Thomas Dugé,Franziska Liesecke RNA-seq quantification from published data. **protocols.io** dx.doi.org/10.17504/protocols.io.qxhdxj6
Published: 13 Jun 2018

Protocol

Step 1.

Identify RNA-seq runs of interests and get a DRR/ERR/SRR accession number.
In a bash script, assign the accession number to the $acc_sra variable.

SOFTWARE PACKAGE (LINUX)

Prozilla, 2.0.4
Kalum Somaratna
https://github.com/totosugito/prozilla-2.0.4

cmd COMMAND (LINUX)

```
                if [ ${#acc_sra} -eq 9 ]; then
                        proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/${acc_sra}/${acc_sra}_1.fastq.gz -
P $acc_wd
                         gunzip ${acc_sra}_1.fastq.gz
                        proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/${acc_sra}/${acc_sra}_2.fastq.gz -
P $acc_wd
                        gunzip ${acc_sra}_2.fastq.gz
                                else
                                if [ ${#acc_sra} -eq 10 ]; then
                                        proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/00${acc_sra:9:9}/${acc_sra}/${acc_sra}_
1.fastq.gz -P $acc_wd
                                        gunzip ${acc_sra}_1.fastq.gz
                                        proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/00${acc_sra:9:9}/${acc_sra}/${acc_sra}_
2.fastq.gz -P $acc_wd
                                        gunzip ${acc_sra}_2.fastq.gz
                                        else
                                        if [ ${#acc_sra} -eq 11 ]; then
                                                proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/0${acc_sra:9:10}/${acc_sra}/${acc_sra}_
1.fastq.gz -P $acc_wd
                                                gunzip ${acc_sra}_1.fastq.gz
                                                proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/0${acc_sra:9:10}/${acc_sra}/${acc_sra}_
```

```
2.fastq.gz -P $acc_wd
                                                gunzip ${acc_sra}_2.fastq.gz
                                                else
                                                        proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/${acc_sra:9:11}/${acc_sra}/${acc_sra}_1
.fastq.gz -P $acc_wd
                                                        gunzip ${acc_sra}_1.fastq.gz
                                                        proz -k=8 --force -r --no-
curses ftp.sra.ebi.ac.uk/vol1/fastq/${acc_sra:0:6}/${acc_sra:9:11}/${acc_sra}/${acc_sra}_2
.fastq.gz -P $acc_wd
                                                        gunzip ${acc_sra}_2.fastq.gz
                                                fi
                                fi


                        fi
```

use Prozilla to download data from the EBI

ANNOTATIONS

Thomas Dugé 13 Jun 2018

k option in prozilla indicates the number of CPUs

Step 2.

Trim reads with Trimmomatic. Assign to $adaptfile variable the appropriate file name containing primer sequences (found in the Trimmomatic folder).

SOFTWARE PACKAGE (LINUX)

Trimmomatic, 0.32

USADELLAB

http://www.usadellab.org/cms/?page=trimmomatic

cmd COMMAND (LINUX)

```
java -jar /home/tduge/trinity/Trimmomatic-0.32/trimmomatic-0.32.jar PE -
threads $thread_number ${acc_sra}_1.fastq ${acc_sra}_2.fastq ${acc_sra}_1P.fq ${acc_sra}_1
U.fq ${acc_sra}_2P.fq ${acc_sra}_2U.fq ILLUMINACLIP:$adaptfile:2:30:10 LEADING:3 TRAILING:
3 SLIDINGWINDOW:4:15 MINLEN:36
```

Trim paired RNA-seq reads with Trimmomatic

Step 3.

Quantify transcript abundance against a reference transcriptome assembly.

Assign the reference transcriptome to the $sid variable (the fasta assembly may be downloaded in Ensembl for example).

SOFTWARE PACKAGE (LINUX)

Salmon, 0.8.2

Rob Patro

https://combine-lab.github.io/salmon/

cmd COMMAND (LINUX)

```
forward=${acc_sra}_1P.fq
reverse=${acc_sra}_2P.fq

salmon index -t $assembly -i ${assembly}_quasi_index --type quasi -k 31
salmon quant -i $sid -l A -p $thread_number --useVBOpt -1 $forward  -2  $reverse -
o ${acc_sra}_transcripts_quant_quasi_vbo --seqBias
```

Prepare reference index and quantify transcript abundance

ANNOTATIONS

Thomas Dugé 13 Jun 2018

assembly and quantification may be switch to the lightweight-alignment (SMEM), please read the doc @ http://salmon.readthedocs.io/en/latest/salmon.html#quasi-mapping-based-mode-including-lightweight-alignment