December 2014

RNA-Seq de novo assembly training session Day 2 hands-on

Useful links:

Assemblathon

An offshoot of the Genome 10K project, and primarily organized by the UC Davis Genome Center, Assemblathons are contests to assess state-of-the-art methods in the field of genome assembly

CD-HIT

CD-HIT is a very widely used program for clustering and comparing protein or nucleotide sequences. CD-HIT was originally developed by Dr. Weizhong Li.

TGICL

This package automates clustering and assembly of a large EST/mRNA dataset. The clustering is performed by a slightly modified version of NCBI's megablast, and the resulting clusters are then assembled using CAP3 assembly program.

Oases

Oases is a de novo transcriptome assembler designed to produce transcripts from short read sequencing technologies, such as Illumina, SOLiD, or 454 in the absence of any genomic assembly.

Trinity

Trinity, developed at the Broad Institute and the Hebrew University of Jerusalem, represents a novel method for the efficient and robust de novo reconstruction of transcriptomes from RNA-seq data.

BWA

BWA is a software package for mapping low-divergent sequences against a large reference genome, such as the human genome. It consists of three algorithms: BWA-backtrack, BWA-SW and BWA-MEM.

Samtools

SAMTools provide various utilities for manipulating alignments in the SAM format, including sorting, merging, indexing and generating alignments in a per-position format.

IGV

The **Integrative Genomics Viewer (IGV)** is a high-performance visualization tool for interactive exploration of large, integrated genomic datasets.

Training session aims:

This training session provides you the manipulation of some *de novo* assemblers.

Data used in the exercises can be found at:
http://genoweb.toulouse.inra.fr/~formation/RNASeq_de_novo/Assembly

Exercise n°1: Assembly quality assessment

For each of the fasta file from the directory exercise_1:

- compute generic metrics using the assemblathon statistics script.
- draw the contig length histogram using the python **`length_histogram.py`** script.
- compute the realignment mapping rates (mapped and paired).
- does one of the assemblies seem obviously better than the others?

Exercise n°2: Assembly using Velvet/Oases

Assemble reads from runs **`ERR145651_t`** and **`ERR145651_t_norm`** inside exercise_2 directory using Velvet/Oases with the following parameters (assemble runs separately):

- k-mers list: 29, 37, 45, 53, 61, 69
- -min_contig_lgth = 200 for velvetg

Use the *_70_LONG command versions (velveth_70_LONG…).

Job resources reservation: **`-l mem=8G,h_vmem=32G`**

Locate the output contigs fasta file (named **`transcripts.fa`** inside the **`oases -merge`** output directories) and:

- compute the realignment mapping rates (mapped and paired).
- Blat contigs to `Danio rerio chr3` and extract the best blat hit (in psl format).
- Exonerate Danio rerio proteins to contigs.

Start IGV and compare assemblies of the two runs. Load following files:

- Genome -> Load genome from file -> `Danio rerio chr3` fasta file
- File -> Load from file :
    - `Danio rerio chr3` GTF file
    - `ERR145651_t_vs_genome` BAM file
    - `ERR145651_t_norm_vs_genome` BAM file
    - `ERR145651_t_vs_genome` TDF file
    - `ERR145651_t_norm_vs_genome` TDF file
    - `ERR145651_t` best blat hits versus genome
    - `ERR145651_t_norm` best blat hits versus genome

igv_exercise_2.xml

Locate particular regions:

- Transcripts correctly assembled using one run and not the other
- All isoforms of transcripts correctly or not correctly assembled

- Contigs found inside UTRs
- Contigs found inside introns
- Transcripts not correctly assembled whereas reads coverage seems sufficient

Run ERR145651_t_norm is a normalized version of the ERR145651_t run. What are the normalization main effects on the assembly?

IGV Tips:

- Once all files have been load and tracks correctly formatted, don't forget to save your session (File -> Save sessions)
- Use the Region -> Region navigator tool to store particular regions

Exercise n°3: Assembly using Trinity

Assemble reads from runs **ERR145651_t** and **ERR145651_s** inside exercise_3 directory using Trinity with the following parameters:

- number of CPUs = 4
- memory = 64G

Job resources reservation: **-l mem=8G,h_vmem=32G -pe parallel_smp 4**

Locate the output contigs fasta file (named **Trinity.fasta** inside the output directories) and:

- compute the realignment mapping rates (mapped and paired).
- Blat contigs to Danio rerio chr3 and extract the best blat hit (in psl format).
- Exonerate Danio rerio proteins to contigs.

Start IGV and compare assemblies of the two runs. Load following files:

- Genome -> Load genome from file -> Danio rerio chr3 fasta file
- File -> Load from file :
    - Danio rerio chr3 GTF file
    - ERR145651_t_vs_genome BAM file
    - ERR145651_s_vs_genome BAM file
    - ERR145651_t_vs_genome TDF file
    - ERR145651_s_vs_genome TDF file
    - ERR145651_t best blat hits versus genome
    - ERR145651_s best blat hits versus genome

igv_exercise_3.xml

Locate particular regions.
Compare mapping rates between ERR145651_t assembled with Trinity, ERR145651_t assembled with oases in exercise n°2 and ERR145651_t_norm assembled with Oases

also in exercise n°2. What about the mapping rates differences? Where would they come from?

For further questions :

- e-mail : support.genopole@toulouse.inra.fr .
- You can check the FAQ of the genotoul website: http://bioinfo.genotoul.fr/index.php?id=11 .
- Using the following link, you can have more information about the other training sessions provided by BIOINFO GENOTOUL: http://bioinfo.genotoul.fr/index.php?id=10.