Docker image: rMATS-turbo-0.1

Prerequisite

- Docker

Software installed in this image

- Operating system: Debian GNU/Linux 8 (jessie)
- gcc version 4.9.2 (Debian 4.9.2-10)
- STAR-2.5.2b
- Python 2.7.12
    - Cython (0.25.2)
    - numpy (1.12.0)
- libblas-dev 1.2.20110419-10
- liblapack-dev 3.5.0-4
- libgsl0ldbl 1.16+dfsg-2

install the image

```
docker load -i rmats-turbo-0.1.tar
```

run the image

```
docker run rmats:turbo01 [options]
```

RMATS USAGE

About

rMATS-turbo is the C/Cython version of rMATS (refer to http://rnaseq-mats.sourceforge.net/user_guide.htm). The main difference between rMATS-turbo and rMATS is speed and space usage. The speed of rMATS-turbo is 100 times faster and the output file is 1000 times smaller than rMATS. These advantages make analysis and storage of large scale dataset easy and convenient.

	Counting part	Statistical part
Speed (C/Cython version vs Python version)	20~100 times faster (one thread)	300 times faster (6 threads)
Storage usage (C/Cython version vs Python version)	1000 times smaller	-

Usage

```
docker run rmats:turbo01 -h

usage: usage: rmats.py [options] arg1 arg2

optional arguments:
      -h, --help            show this help message and exit
      --version             Version.
      --gtf GTF             An annotation of genes and transcripts in GTF format.
      --b1 B1               BAM configuration file.
      --b2 B2               BAM configuration file.
      --s1 S1               FASTQ configuration file.
      --s2 S2               FASTQ configuration file.
      --od OD               output folder of post step.
      -t {paired,single}    readtype, single or paired.
      --libType {fr-unstranded,fr-firststrand,fr-secondstrand}
                            Library type. Default is unstranded (fr-unstranded).
                            Use fr-firststrand or fr-secondstrand for strand-
                            specific data.
      --readLength READLENGTH
                            The length of each read.
      --anchorLength ANCHORLENGTH
                            The anchor length. (default is 1.)
      --tophatAnchor TOPHATANCHOR
                            The "anchor length" or "overhang length" used in the
                            aligner. At least “anchor length” NT must be
                            mapped to each end of a given junction. The default is
                            1. (This parameter applies only if using fastq).
      --bi BINDEX           The folder name of the STAR binary indexes (i.e., the
                            name of the folder that contains SA file). For
                            example, use ~/STARindex/hg19 for hg19. (Only if using
                            fastq)
      --nthread NTHREAD     The number of thread. The optimal number of thread
                            should be equal to the number of CPU core.
      --tstat TSTAT         the number of thread for statistical model.
      --cstat CSTAT         The cutoff splicing difference. The cutoff used in the
                            null hypothesis test for differential splicing. The
                            default is 0.0001 for 0.01% difference. Valid: 0 ≤
                            cutoff < 1.
      --statoff             Turn statistical analysis off.
```

Output

--od read count generated by the post step:

- fromGTF.AS_Event.txt: all possible alternative splicing (AS) events derived from GTF and RNA.
- JC.raw.input.AS_Event.txt evaluates splicing with only reads that span splicing junctions
    - IJC*SAMPLE*1: inclusion junction counts for SAMPLE_1, replicates are separated by comma
    - SJC*SAMPLE*1: skipping junction counts for SAMPLE_1, replicates are separated by comma
    - IJC*SAMPLE*2: inclusion junction counts for SAMPLE_2, replicates are separated by comma
    - SJC*SAMPLE*2: skipping junction counts for SAMPLE_2, replicates are separated by comma
    - IncFormLen: length of inclusion form, used for normalization
    - SkipFormLen: length of skipping form, used for normalization
- JCEC.raw.input.AS_Event.txt evaluates splicing with reads that span splicing junctions and reads on target (striped

regions on home page figure)

    - IC*SAMPLE*1: inclusion counts for SAMPLE_1, replicates are separated by comma
    - SC*SAMPLE*1: skipping counts for SAMPLE_1, replicates are separated by comma
    - IC*SAMPLE*2: inclusion counts for SAMPLE_2, replicates are separated by comma
    - SC*SAMPLE*2: skipping counts for SAMPLE_2, replicates are separated by comma
    - IncFormLen: length of inclusion form, used for normalization
    - SkipFormLen: length of skipping form, used for normalization
- AS_Event.MATS.JC.txt evaluates splicing with only reads that span splicing junctions

    - IC*SAMPLE*1: inclusion counts for SAMPLE_1, replicates are separated by comma
    - SC*SAMPLE*1: skipping counts for SAMPLE_1, replicates are separated by comma
    - IC*SAMPLE*2: inclusion counts for SAMPLE_2, replicates are separated by comma
    - SC*SAMPLE*2: skipping counts for SAMPLE_2, replicates are separated by comma
- AS_Event.MATS.JCEC.txt evaluates splicing with reads that span splicing junctions and reads on target (striped regions on home page figure)

    - IC*SAMPLE*1: inclusion counts for SAMPLE_1, replicates are separated by comma
    - SC*SAMPLE*1: skipping counts for SAMPLE_1, replicates are separated by comma
    - IC*SAMPLE*2: inclusion counts for SAMPLE_2, replicates are separated by comma
    - SC*SAMPLE*2: skipping counts for SAMPLE_2, replicates are separated by comma
- Important columns contained in output files above

    - IncFormLen: length of inclusion form, used for normalization
    - SkipFormLen: length of skipping form, used for normalization
    - P-Value: (The meaning of p value???)
    - FDR: (The meaning of FDR???)
    - IncLevel1: inclusion level for SAMPLE_1 replicates (comma separated) calculated from normalized counts
    - IncLevel2: inclusion level for SAMPLE_2 replicates (comma separated) calculated from normalized counts
    - IncLevelDifference: average(IncLevel1) - average(IncLevel2)
- bamX_Y STAR mapping result.

How to transfer data to docker image's file system.

Docker has it's own file system, called Union File System. We're not going to dig into these concepts. What we're going to do is to learn how we can manage data inside and between our Docker containers.

Suppose our BAM files and GTF files are stored in /yourdatafolder, and we're going to use rMATS-turbo to analyze them. Docker can't access these file for security reason. In order to make these file visible to Docker, we have to use option -v (https://docs.docker.com/engine/tutorials/dockervolumes/#data-volumes). This option will mount our local folder to docker's file system, and retrieve output from docker.

Note that, after mounting our folder to docker, docker can read this folder, and it can also write output file to this folder.

Examples

Suppose we have 4 samples in /yourdatafolder.

```
$ ls /yourdatafolder:
- b1.txt
- b2.txt
- 5.gtf
- 1.bam
- 2.bam
- 3.bam
- 4.bam

$ cat b1.txt:
/data/1.bam,/data/2.bam

$ cat b2.txt:
/data/3.bam,/data/4.bam

docker run -v /yourdatafolder:/data rmats:turbo01 --b1 /data/b1.txt \
    --b2 /data/b2.txt --gtf /data/5.gtf --od /data/output -t paired \
    --nthread 4 --readLength 101 --anchorLength 1
```

This command mounts the host directory, /yourdatafolder, into the container at /data. If the path /data already exists inside the container's image, the /yourdatafolder mount overlays but does not remove the pre-existing content. Once the mount is removed, the content is accessible again. This is consistent with the expected behavior of the mount command.

Accordingly, the absolute path of file should be adjusted. (e.g. b1.txt, 5.gtf, 2.bam, etc. changed to /data/b1.txt, /data/5.gtf, /data/2.bam, etc.)

Important note: The output folder /data/output will be written to /yourdatafolder/output.