==========

Mixture model for strains reconstruction – Windows Version

Author: XIN LI

Date: 04/23/2020

==========

######Working OS##################

Our model is working on docker in Windows Operating System.

Virtual Environment

Some of our dependencies are only workable under Linux OS. So if you really want to run in under Windows OS, you can run inside a docker inside windows OS. We already provided a Dockerfile inside mixture directory.

(1) Install docker in windows:

https://docs.docker.com/docker-for-windows/install/

(2) When you have installed docker inside your windows OS, then open a command prompt in mixture model directory, then build the image:

docker build -t mixture .

Other possible method

1. Virtual Box: Install Virtual box under windows OS, and install a virtual machine with ubuntu OS inside. And see linux version of our manual

2.Windows subsystem for Linux: Since windows10, Microsoft provide a subsystem inside windows that allowed you run Linux OS as terminal inside Windows 10 OS.

https://docs.microsoft.com/en-us/windows/wsl/install-win10

Parameters for Test Mixture model

(1) Open a command prompt inside your windows OS, then run following command to into docker terminal:

docker run -v D:\tmp\windows:/mixutre -it mixture bash

NOTE: D:\tmp\windows change it to the root directory of your downloaded our mixture model directory (where the mixture_model.py is).

(2) Then you actually go into root directory of a Linux OS, the mixture model is saved into the directory /mixture/, go into that directory by:

cd /mixture

(3)Run our mixture model insider your mixture directory:

python ./mixture_model.py

--sample_name sample_name: Given a unique sample name for running on samples

--genome_len genomeLength: the corresponding genome length

--genome_name genomeName: the genome name(name inside FASTA file)

--genome_file_loc genomeLoc: genome FASTA file location. Absolute file location is better

--bam_file bam_file: the sorted bam file which have been mapped to the reference genome.

--res_dir directory_name: the result directory name

==========

Run for real-world data for Mixture model

(1)You need to map your windows data directory into docker Linux directory when you run docker, like below:

docker run -v D:\tmp\data:/mixture/data -it mixture bash

D:\tmp\data is the data directory inside your windows OS, when you run this command, you will find your data under Windows OS are actually stored inside the /mixture/data of docker Linux OS. You can also set up other destination directory, like /data/. Just remember, you need to find your data inside the directory when you get into docker terminal. When you set your –res_dir parameter, you should also set up the inside the /mixture/data.

==========

Example

(1) If you install by docker, then open a command prompt inside your windows OS, then run following command to into docker terminal:

docker run -v D:\tmp\windows:/mixture -it mixture bash

D:\tmp\windows -> It is the local location in Windows

:/mixture -> it is mapped location inside Linux. You can changed this directory, but you also need to change directory below.

Then you actually go into root directory of a Linux OS, the mixture model is saved into the directory /mixture/, go into that directory by:

cd /mixture

(2)The test data has been saved in the directory of example_test_data, so you can just navigate into directory where mixture_model.py located. And use below command to run example data. The result will save into directory test_result.

python mixture_model.py --sample_name test --genome_len 1853160 --genome_name NC_009515.1 --genome_file_loc ./example_test_data/GCF_000016525.1_ASM1652v1_genomic.fna --bam_file ./example_test_data/test_sorted.bam --res_dir ./test_result

==========

Preprocessing

Since MixtureS require bam format input, you had better input the sorted Bam file as input. However, if you only have FastQ format data, you can get a hint from our common preprocessing pipeline. But please check your experimental specification before applying our preprocessing step, because our preprocessing may be not suitable for your specific experiments. So you may need to adjust our preprocessing script based on your own experiments or trim read protocol, like single-end read, trim customized bar code or others.

(1) Use docker like above
(2) Sample command:

python preprocessing.py --sample_name test --pair1 ./example_test_data/test.read1.fastq --pair2 ./example_test_data/test.read2.fastq --process 6 --genome_name ./example_test_data/GCF_000016525.1_ASM1652v1_genomic.fna --res_dir ./test_res_data

--sample_name: sample name

--pair1: forward read for FastQ

--pair2: reverse read for FastQ

--process: # of processor to run

--genome_name: reference genome

--res_dir : result directory

==========

Interpret results

1.sampleName_strains file: This is the final prediction of strains after combining k-mer results. It includes coverage with corresponding polymorphic sites. In the example below, there are two strains predicted. Each strain starts with '>' and its corresponding coverage. From next row, they are the

polymorphic sites of above strain. Each row is represented by allele and its position. In the example below, two strains with coverage 10.402205435195318 and 23.599084994628157. The first strain has six polymorphic sites, and the second strain has 5 polymorphic sites.

output example:

>10.402205435195318

C,807324

A,260641

G,869404

T,143273

G,902916

A,754450

>23.599084994628157

G,576282

G,1500940

C,162805

T,924389

C,1722675

2.Polymorphic sites:

First column is location, and followed by frequency of A, C, G ,T. For example, in the first row below, the location is 6696 with frequency of C 6 and frequency of G 175.

6696	0	9	175	0
7148	161	0	6	0
7589	0	0	4	203
10403	0	203	4	0

==========

Contact Information

Please do not hesitate to reach out to me if you have questions.

Xin Li (xli042@knights.ucf.edu)

Haiyan Nancy Hu (haihu@cs.ucf.edu)

Xiaoman Shawn Li (xiaoman@mail.ucf.edu)