NCHC for graphKIR pipeline

Created time: April 22, 2023 3:44 PM Formula: April 22 2023

create a branch and upload new code (Don’t change too much)

Require

- login NCHC and create a user
- module load anaconda

```
module load pkg/Anaconda3
```

- create a virtual env : `ENV_NAME`
- index

Usage

- Download Graph-KIR pre-built index

```
git clone https://github.com/linnil1/KIR_graph
cd KIR_graph
pip install .
graphkir --help
```

Step 1: Run a test sample locally and generate an index. Step 2: Build the index based on the test sample. Step 3: Upload the index to the server. Alternatively, Step 1: Run a test sample on the local network, generating an index (with the risk of being interrupted by the TWCC). Step 2: Build the index based on the test sample.

- run for two sample :
    - route : `/data`
    - sample : `example_name`

```
graphkir \
    --thread 2 \
    --r1 data/example_name.read.1.fq \
    --r2 data/example_name.read.2.fq \
    --index-folder example_index \
    --output-folder example_data \
    --output-cohort-name example_data/cohort
        --engine local
```

How to submit job by slurm

Required some `.sh`

- summit job :
  - `submit_graphkir.sh`

```
#!/bin/bash
wkdir=your workdir
SampleList=${wkdir}/sampleID.txt
PIPELINE=${wkdir}/run_graphkir_single.sh #.read1.fq.gz
DAY=$(date +%Y%m%d)

mkdir -p ${wkdir}/log
while read -r ID; do
    cd ${wkdir}
    rsync ${PIPELINE} ./log/${DAY}graphkir_${ID}.sh

    #change ID to SAMPLE_NAME variable, therefore we can read this variable in other files
    sed -i "s|SAMPLE_NAME|${ID}|g" ./log/${DAY}graphkir_${ID}.sh

        #submit your job in TWCC
    sbatch ./log/${DAY}graphkir_${ID}.sh

    # add sampleID to env
    echo SampleID=${ID}
    echo $PWD
    sleep 3s
    cd ${wkdir}
done <${SampleList}
```

- run job :
  - `run_graphkir_single.sh`

```
#!/bin/bash
#SBATCH -A MST109178            #Account name/project number
#SBATCH -J commend        #job name (上限：250*time limit, 120*, 80, 60, 30)
#SBATCH -p ngs53G            # Partition Name(ngs7G,ngs13G,ngs26G,ngs53G,ngs92G,ngs186G,ngs372G)
#SBATCH -c 8                # 使用的core數 請參考Queue資源設定 (1,2,4,8,14,28,56)
#SBATCH --mem=53g            # 使用的記憶體量 請參考Queue資源設定
#SBATCH -o /log/out_SAMPLE_NAME.log    # Path to the standard output file
#SBATCH -e /log/err_SAMPLE_NAME.log    # Path to the standard error ouput file
#SBATCH --mail-user=r08424026@g.ntu.edu.tw    # email
#SBATCH --mail-type=FAIL,END         #Mail events(NONE,BEGIN,END,FAIL,ALL)

SampleID=SAMPLE_NAME

cd workdir # please modify and put on your work directory route

module load pkg/Anaconda3
conda activate ENV_NAME

module load biology/BWA/0.7.17
module load biology/HISAT2/2.2.1
module load biology/Samtools/1.15.1
module load libs/singularity/3.10.2

date
graphkir \
    --thread 8 \
    --r1 hprc_data/${SampleID}.read.1.fq.gz \
    --r2 hprc_data/${SampleID}.read.2.fq.gz \
    --engine local \
    --index-folder example_index \
    --output-folder single_sample
```

- total sample ID file :
    - `sampleID.txt`

```
HG002
HG003
HG004
```

You need to add an extra newline character at the end of your .txt file to make the while loop in your `.sh` script work correctly. The while loop reads each line of the text file, and without the extra newline character, the last line may not be read correctly. By adding an extra newline character at the end of the file, you ensure that the last line is properly read and processed by the while loop.