GATK tutorial using docker container

Description

GATK: GATK4 toolkit offers a wide variety of tools with a primary focus on variant discovery and genotyping. The content on this page is borrowed from GATK webpages/courses. To get familiar with GATK tools, you can read the following:

- Read the overview of GATK4 : https://software.broadinstitute.org/gatk/gatk4
- Quick start guide: https://software.broadinstitute.org/gatk/documentation/quickstart
- Presentation materials: https://software.broadinstitute.org/gatk/documentation/presentations
- Best Practices workflows : https://software.broadinstitute.org/gatk/best-practices/

Download GATK container from Dockerhub

```
docker pull broadinstitute/gatk:latest
```

or with some specific-version information

```
docker pull broadinstitute/gatk:4.0.11.0
```

Run GATK container

```
docker run -it broadinstitute/gatk:latest
```

Run a GATK command inside the container

```
./gatk --list
```

The general format for GATK commands is

gatk ToolName [tool args]

Exit from the container

```
ctrl+p then ctrl+q
```

Download example data to local folder

wget https://object.pouta.csc.fi/Softwares/data.zip

Start running gatk container and mount the location of the data bundle inside the docker container

```
docker run -v /path/gatk_data:/gatk/data -it broadinstitute/gatk:latest
```

Get usage information for a GATK command

```
gatk HaplotypeCaller --help
```

Run HaplotypeCaller

```
gatk HaplotypeCaller -R data/ref/ref.fasta -I data/bams/mother.bam \
-O data/sandbox/variants.vcf
```

Add JVM options to the command

```
gatk --java-options "-Xmx4G" HaplotypeCaller \
-R data/ref/ref.fasta -I data/bams/mother.bam \
-O data/sandbox/variants.vcf
```

Run GVCF workflow tools using HaplotypeCaller, GenomicsDBImport and then GenotypeGVCFs to perform joint calling on multiple input samples.

Run HaplotypeCaller on three input bams (mother, father, son)

```
gatk HaplotypeCaller -R data/ref/ref.fasta -I data/bams/mother.bam -O
data/sandbox/mother.g.vcf -ERC GVCF

gatk HaplotypeCaller -R data/ref/ref.fasta -I data/bams/father.bam -O
data/sandbox/father.g.vcf -ERC GVCF

gatk HaplotypeCaller -R data/ref/ref.fasta -I data/bams/son.bam -O
data/sandbox/son.g.vcf -ERC GVCF
```

Run GenomicsDBImport on three GVCFs to consolidate

```
gatk GenomicsDBImport -V data/sandbox/mother.g.vcf \
-V data/sandbox/father.g.vcf \
-V data/sandbox/son.g.vcf --genomicsdb-workspace-path \
data/sandbox/trio.gdb_workspace --intervals 20
```

Alternatively, use CombinedGVCFs command as an alternative to GenomicsDBImport

```
gatk CombineGVCFs -R data/ref/ref.fasta \
-V data/sandbox/father.g.vcf \
-V data/sandbox/mother.g.vcf -V data/sandbox/son.g.vcf \
-O data/sandbox/combine_trio_variants.vcf
```

Run GenotypeGVCFs on the GDB workspace to produce final multisample VCF

```
gatk GenotypeGVCFs -R data/ref/ref.fasta \
-V gendb://data/sandbox/trio.gdb_workspace \
-G StandardAnnotation -O data/sandbox/trio_variants.vcf
```

Run a command with local Spark multithreading

```
gatk --java-options "-Xmx6G" MarkDuplicatesSpark -R data/ref/ref.fasta \
-I data/bams/mother.bam -O data/sandbox/mother_dedup.bam \
-M data/sandbox/metrics.txt -- --spark-master local[*]
```

delete all containers

```
sudo docker rm `sudo docker ps --no-trunc -aq`
```

```
docker ps -a | grep 'weeks ago' | awk '{print $1}' | xargs docker rm
```

Delete stopped containers

```
docker rm -v $(docker ps -a -q -f status=exited)
```

From inside container’s command prompt, detach and return to the host’s prompt.

```
ctrl+p then ctrl+q
```

Getting Docker Container's IP Address from host machine:

```
docker inspect --format '{{ .NetworkSettings.IPAddress }}' $(docker ps -
q)
```

Useful to know

docker logs - gets logs from container
docker events - gets events from container
docker port - shows public facing port of container
docker top - shows running processes in container
docker stats - shows containers' resource usage statistics
docker diff - shows changed files in the container's FS

add user to docker group and type ID /check UID

```
sudo usermod -aG docker $user
```