**<< Strelka2 >>**

== 00\_human\_b37\_script\_Strelka2.sh

normal=$1

tumor=$2

sampleName=$3

set -e

/home/users/yssong/tools/Strelka2/strelka-2.9.9.centos6\_x86\_64/bin/configureStrelkaSomaticWorkflow.py --normalBam $1 --tumorBam $2 --ref /home/users/yssong/99\_reference/human/GRCh37/human\_g1k\_v37.fasta --runDir $3 &>$3.strelka2.out &&

$3/runWorkflow.py -m local -j 4 &> $3.strelka2.runWorkflow.out

== 00\_1\_run\_strelka2\_2.sh

sh 00\_human\_b37\_script\_Strelka2.sh /home/users/team\_projects/uveal\_melanoma/02\_bam/1-17-Blood.s.md.ir.br.bam /home/users/team\_projects/uveal\_melanoma/02\_bam/1-15-Tumor.s.md.ir.br.bam 1-15-Tumor

sh 00\_human\_b37\_script\_Strelka2.sh /home/users/team\_projects/uveal\_melanoma/02\_bam/3-29-Blood.s.md.ir.br.bam /home/users/team\_projects/uveal\_melanoma/02\_bam/3-31-Tumor.s.md.ir.br.bam 3-31-Tumor

sh 00\_human\_b37\_script\_Strelka2.sh /home/users/team\_projects/uveal\_melanoma/02\_bam/4-42-Blood.s.md.ir.br.bam /home/users/team\_projects/uveal\_melanoma/02\_bam/4-41-Tumor.s.md.ir.br.bam 4-41-Tumor

**<< Varscan2 >>**

== 00\_human\_b37\_varscan.sh

pwd\_mpileup=$1

tumor=$2

normal=$3

java -XX:+UseParallelGC -XX:ParallelGCThreads=1 -Xmx12g -jar /home/users/tools/varscan2.4.2/VarScan.v2.4.2.jar somatic $1/$3.mpileup $1/$2.mpileup $2.varscan --min-var-freq 0.01 --output-vcf 1 &> $2.varscan.out

mv $2.varscan.out $2.varscan.success

== 00\_1\_run\_varscan\_191206\_human.sh

sh 00\_human\_b37\_varscan.sh /home/users/yssong/uveal\_melanoma/03\_mpileup 1-17-Tumor 1-15-Blood

**<< mpileup >>** : varscan2, sequenza는 미리 mpileup by sequenza을 진행해야함.

== 00\_human\_b37\_script\_mpileup\_prev.sh

sampleID=$1

samtools mpileup -B -Q 20 -q 20 -f /home/users/yssong/99\_reference/human/GRCh37/human\_g1k\_v37.fasta /home/users/team\_projects/uveal\_melanoma/02\_bam/$1.s.md.ir.br.bam -o $1.mpileup &> $1.mpileup.out

== 00\_1\_run\_mpileup.sh

sh 00\_human\_b37\_script\_mpileup.sh 1-17-Tumor

sh 00\_human\_b37\_script\_mpileup.sh 1-15-Blood

**<< Delly >>**

bam\_pwd=$1

tumor=$2

normal=$3

/home/users/tools/delly-0.7.6/delly/src/delly call -t DEL -q 15 -o $tumor.DEL.bcf -g /home/users/yssong/99\_reference/human/GRCh37/human\_g1k\_v37.fasta $bam\_pwd/$tumor.s.md.bam $bam\_pwd/$normal.s.md.bam &> $tumor.DEL.out &

/home/users/tools/delly-0.7.6/delly/src/delly call -t INS -q 15 -o $tumor.INS.bcf -g /home/users/yssong/99\_reference/human/GRCh37/human\_g1k\_v37.fasta $bam\_pwd/$tumor.s.md.bam $bam\_pwd/$normal.s.md.bam &> $tumor.INS.out &

/home/users/tools/delly-0.7.6/delly/src/delly call -t DUP -q 15 -o $tumor.DUP.bcf -g /home/users/yssong/99\_reference/human/GRCh37/human\_g1k\_v37.fasta $bam\_pwd/$tumor.s.md.bam $bam\_pwd/$normal.s.md.bam &> $tumor.DUP.out &

/home/users/tools/delly-0.7.6/delly/src/delly call -t INV -q 15 -o $tumor.INV.bcf -g /home/users/yssong/99\_reference/human/GRCh37/human\_g1k\_v37.fasta $bam\_pwd/$tumor.s.md.bam $bam\_pwd/$normal.s.md.bam &> $tumor.INV.out &

/home/users/tools/delly-0.7.6/delly/src/delly call -t TRA -q 15 -o $tumor.TRA.bcf -g /home/users/yssong/99\_reference/human/GRCh37/human\_g1k\_v37.fasta $bam\_pwd/$tumor.s.md.bam $bam\_pwd/$normal.s.md.bam &> $tumor.TRA.out

wait

#echo finish

#echo "start SV annotation from bcf combine"

(bcftools concat -a -O v -o $tumor.delly.vcf $tumor.DEL.bcf $tumor.DUP.bcf $tumor.INS.bcf $tumor.INV.bcf $tumor.TRA.bcf) &> $2.concat.out

#echo "done"

**<< Sequenza >>**

#!/bin/bash

set -e

mpileup\_pwd=$1

sampleName=$2

normalPileup=$3

#create seqz file from the two mpileups

if false; then

echo bam2seqz

sequenza-utils bam2seqz -gc /home/users/yssong/09\_uveal\_melanoma/04\_sequenza/gc50base.hg37.wig.gz -n $1/$normalPileup.mpileup -t $1/$sampleName.mpileup -p -o $sampleName.seqz > $sampleName.bam2seqz.out 2>&1

#binning

echo binning

sequenza-utils seqz\_binning --window 100 --seqz $sampleName.seqz -o $sampleName.comp.seqz > $sampleName.binnnig.out 2>&1

# remove unwanted contigs in reference fasta

echo mtglremove

cat $sampleName.comp.seqz | grep -v 'MT' | grep -v 'GL' | grep -v 'JH' > $sampleName.comp.seqz.rmGLMTJH 2> $sampleName.mtglremove.out

#cat $sampleName.comp.seqz | grep -v 'chrM' | grep -v 'GL' | grep -v 'chrUn'| grep -v 'HLA' | grep -v 'KI' | grep -v 'JH' > $sampleName.comp.seqz.rmGLMTJH 2> $sampleName.mtglremove.out

## gzip

echo gzip

gzip $sampleName.comp.seqz.rmGLMTJH > $sampleName.gzip.out 2>&1

echo done

#cleanup

rm $sampleName.seqz $sampleName.comp.seqz

fi

# further analysis

#Rscript /home/users/yssong/09\_uveal\_melanoma/04\_sequenza/11\_Rscript\_sequenzawithpurity.R $sampleName.comp.seqz.rmGLMTJH.gz $sampleName-withpurity 1 2 &> $sampleName.purity.Rscript.out

/home/users/kjyi/miniconda3/bin/Rscript 12\_Rscript\_sequenza\_6args\_nopurityassign.R $sampleName.comp.seqz.rmGLMTJH.gz $sampleName 0 0 hg19 XX &> $sampleName.Rscript.out

#Rscript 11\_Rscript\_sequenza\_6args.R $sampleName.comp.seqz.rmGLMTJH.gz $sampleName 1 2 hg19 XX &> $sampleName.Rscript.out

<< Sequenza wiggle 생성 >>

sequenza-utils gc\_wiggle -w 50 -o gc50base.ucsc.wig -f /home/users/yssong/99\_reference/human/hg19\_UCSC/ucsc.hg19.fasta > gc50basewiggle.ucsc.out 2>&1 &&

gzip gc50base.ucsc.wig &> gc50.ucsc.gzip.out