Caderno Computacional 5/6: Identificação de Variantes Genéticas (Variant Call) de SARS-CoV-2 com arquivos FASTQ

Gepoliano Chaves

Outubro, 2020

Contents

Introdução

Neste notebook, identificamos variantes de SARS-CoV-2 usando sratools e a base de dados Gene Expression Omnibus (GEo) para extrair polimorfismos de arquivos FASTQ. A instalação de SRATools permite o download direto de arquivos de expressao genica a partir da base GEO. GEO é um banco de dados de expressão gênica gerenciado pelo *National Center for Biotechnology Information* (NCBI) dos EUA.

1) Instruçoes para servidor

```
#!/bin/bash
##SBATCH -p 128x24           ## particao a ser usada
#SBATCH --job-name=CV_Variants # nome do trabalho
##SBATCH --mail-type=ALL              # Mail events (NONE, BEGIN, END, FAIL, ALL)
#SBATCH --mail-user=gchaves@ucsc.edu  # Where to send mail
##SBATCH --nodes=10                    # Use one node
##SBATCH --ntasks=10                   # Run a single task
##SBATCH --cpus-per-task=10            # Number of CPU cores per task
##SBATCH --output=CV_Variants     # Standard output and error log
##SBATCH --time=1000
# module load gcc/5.4.0

## Esta linha apresenta a partição a ser usada para executar o script.
#sbatch --partition=128x24 ~/scripts/Signaling_Analysis.sh

## Esta outra linha permite o reconhecimento do ~/.bash_profile e
## consequente acesso aos softwares ai presente:
source ~/.bashrc
```

2) Instalacao SRA-tools e BWA usando Anaconda e Indexamento de Genoma

2.1) Instalar SRA-tools e BWA

SRA-tools é usado para baixar os arquivos FASTQ diretamente da base de dados.

```
conda install -c bioconda sra-tools
## Eh necessario baixar o genoma de referencia de SARS-CoV-2 do Genome Browser da UCSC.
conda install -c bwa ## Preciso verificar este comando depois
```

2.2) Indexamento do Genoma de Referencia

```
bwa index -a bwtsw ~/references/covid_sars/GCF_009858895.2_ASM985889v3_genomic.fna
```

4) Pasta do Projeto e especificacao da Lista GEO

Define o diretório de trabalho ou o diretório onde todos os dados baixados serão armazenados. Faz com que o script de shell funcione com base no loop for, iterando na lista de arquivos FASTQ de GEO. O Nome do projeto define onde os resultados da análise serao armazenados. Dentro do projeto, os mesmos arquivos provenientes da análise da base de dados GEO são armazenados como diretórios. O nome do projeto define o nome do diretório de trabalho onde todos os arquivos resultantes da análise de um arquivo FASTQ GEO, serão armazenados.

```
ProjectDirectory=COVID_BWA_Variant_Call
mkdir ~/$ProjectDirectory

for fastq_file in $(cat ~/$ProjectDirectory/COVID_List_Single_Cell.txt)
do

  FastqDirectory=$fastq_file
  mkdir ~/$ProjectDirectory/$FastqDirectory
```

5) Baixar Arquivos FASTQ do Banco de Dados GEO

Esta parte eh executada utilizando-se o pacote SRA-tools, logo so pode ser feita apos a instalacao do mesmo.

```
  fastq-dump --outdir ~/$ProjectDirectory/$FastqDirectory --split-files $fastq_file
```

6) Definir Genoma de Referencia e localizacao dos arquivos FASTQ

```
  REFERENCE=~/references/covid_sars/GCF_009858895.2_ASM985889v3_genomic.fna
  FastqR1=~/$ProjectDirectory/$FastqDirectory/$fastq_file"_1.fastq"
  FastqR2=~/$ProjectDirectory/$FastqDirectory/$fastq_file"_2.fastq"
```

7) Preparacao para GATK

7.1) Alinhar

```
  bwa mem -M -R '@RG\tID:Sample_W1\tLB:sample_1\tPL:ILLUMINA\tPM:HISEQ\tSM:Sample_W1'
  $REFERENCE $FastaR1 $FastaR2 > ~/$WorkingDirectory/bwa_aligned_reads.sam
  bwa mem -M -R '@RG\tID:SampleCorona\tLB:sample_1\tPL:ILLUMINA\tPM:HISEQ\tSM:SampleCorona'
  $REFERENCE ~/scripts/COVID_Variant_Calling/bahia_file_R1.fastq  >
  ~/$ProjectDirectory/$FastqDirectory/bwa_aligned_reads.sam
```

7.2) Ordenar

```
java -Xmx2g -Djava.io.tmpdir=`pwd`/tmp
java -jar ~/programs/picard-tools-1.140/PicardCommandLine
SortSam INPUT=~/$ProjectDirectory/$FastqDirectory/bwa_aligned_reads.sam
OUTPUT=~/$ProjectDirectory/$FastqDirectory/sorted_reads.bam
SORT_ORDER=coordinate TMP_DIR=`pwd`/tmp
```

7.3) Etapa 3: Coletar Metricas de Alinhamento e de Tamanho das Sequencias

7.3.1) Metricas de Alinhamento

```
java -Xmx2g -Djava.io.tmpdir=`pwd`/tmp
java -jar ~/programs/picard-tools-1.140/PicardCommandLine
CollectAlignmentSummaryMetrics R=$REFERENCE
I=~/$ProjectDirectory/$FastqDirectory/sorted_reads.bam
O=~/$ProjectDirectory/$FastqDirectory/alignment_metrics.txt TMP_DIR=`pwd`/tmp
```

7.3.2) Metricas de Tamanho das Sequencias

```
java -Xmx2g -Djava.io.tmpdir=`pwd`/tmp
java -Xmx2g -Djava.io.tmpdir=`pwd`/tmp -jar
~/programs/picard-tools-1.140/PicardCommandLine CollectInsertSizeMetrics
INPUT=~/$ProjectDirectory/$FastqDirectory/sorted_reads.bam
OUTPUT=~/$ProjectDirectory/$FastqDirectory/insert_metrics.txt
HISTOGRAM_FILE=~/$ProjectDirectory/$FastqDirectory/insert_size_histogram.pdf
TMP_DIR=`pwd`/tmp
```

7.3.3) Metricas de Cobertura das Sequencias

```
   samtools depth -a ~/$ProjectDirectory/$FastqDirectory/sorted_reads.bam >
   ~/$ProjectDirectory/$FastqDirectory/depth_out.txt
```

7.4) Etapa 3: Marcar Duplicados

```
   java -jar ~/programs/picard-tools-1.140/PicardCommandLine MarkDuplicates
   INPUT=~/$ProjectDirectory/$FastqDirectory/sorted_reads.bam
   OUTPUT=~/$ProjectDirectory/$FastqDirectory/dedup_reads.bam
   METRICS_FILE=~/$ProjectDirectory/$FastqDirectory/metrics.txt


#Fixing malformation of bam file
#PicardCommandLine AddOrReplaceReadGroups RGLB=illumina
#RGPL="illumina" RGPU=unit1 RGSM=20 INPUT=~/variant-calling/dedup_reads.bam
#OUTPUT=~/variant-calling/dedup_reads_fixed.bam
```

7.5) Etapa 5: Construir indexamento BAM

```
    java -jar ~/programs/picard-tools-1.140/PicardCommandLine BuildBamIndex
    INPUT=~/$ProjectDirectory/$FastqDirectory/dedup_reads.bam
```

8) Etapa 6: GATK

```
#Step 6: Create Realignment Targets
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar --filter_reads_with_N_cigar
    -T RealignerTargetCreator -R $REFERENCE
    -I ~/$ProjectDirectory/$FastqDirectory/dedup_reads.bam
    -o ~/$ProjectDirectory/$FastqDirectory/realignment_targets.list
```

8.1) Etapa 7: Realinhar Indels

```
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar  --filter_reads_with_N_cigar
    -T IndelRealigner -R $REFERENCE -I ~/$ProjectDirectory/$FastqDirectory/dedup_reads.bam
    -targetIntervals ~/$ProjectDirectory/$FastqDirectory/realignment_targets.list
    -o ~/$ProjectDirectory/$FastqDirectory/realigned_reads.bam
```

8.2) Etapa 7: Identificar variantes (Variant Call)

```
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar  -T HaplotypeCaller -R $REFERENCE
    -I ~/$ProjectDirectory/$FastqDirectory/realigned_reads.bam
    -o ~/$ProjectDirectory/$FastqDirectory/raw_variants.vcf
```

8.3) Etapa 9: Extrair SNPs e Indels

```
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar  -T SelectVariants
    -R $REFERENCE -V ~/$ProjectDirectory/$FastqDirectory/raw_variants.vcf
    -selectType SNP -o ~/$ProjectDirectory/$FastqDirectory/raw_snps.vcf

    java -jar ~/programs/gatk/GenomeAnalysisTK.jar -T SelectVariants
    -R $REFERENCE -V ~/$ProjectDirectory/$FastqDirectory/raw_variants.vcf
    -selectType INDEL -o ~/$ProjectDirectory/$FastqDirectory/raw_indels.vcf
```

8.4) Etapas 10 e 11: Filtrar SNPs e Indels

8.4.1) Filtrar SNPs

```
#Step 10: Filter SNPs
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar -T VariantFiltration -R $REFERENCE
    -V ~/$ProjectDirectory/$FastqDirectory/raw_snps.vcf
    --filterExpression
```

```
    'QD < 2.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 ||
    ReadPosRankSum < -8.0 || SOR > 4.0' --filterName "basic_snp_filter"
    -o ~/$ProjectDirectory/$FastqDirectory/filtered_snps.vcf
```

8.4.2) Filtrar Indels

```
#Step 11: Filter Indels
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar -T VariantFiltration -R $REFERENCE
    -V ~/$ProjectDirectory/$FastqDirectory/raw_indels.vcf
    --filterExpression 'QD < 2.0 || FS > 200.0 || ReadPosRankSum < -20.0 || SOR > 10.0'
    --filterName "basic_indel_filter"
    -o ~/$ProjectDirectory/$FastqDirectory/filtered_indels.vcf
```

8.5) Etapa 12: Recalibracao da nota de Qualidade de Bases #1

```
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar --maximum_cycle_value 35000
    -T BaseRecalibrator -R $REFERENCE
    -I ~/$ProjectDirectory/$FastqDirectory/realigned_reads.bam
    -knownSites ~/$ProjectDirectory/$FastqDirectory/filtered_snps.vcf
    -knownSites ~/$ProjectDirectory/$FastqDirectory/filtered_indels.vcf
    -o ~/$ProjectDirectory/$FastqDirectory/recal_data.table
```

8.6) Etapa 13: Recalibracao da nota de Qualidade de Bases #2

```
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar --maximum_cycle_value 35000
    -T BaseRecalibrator -R $REFERENCE
    -I ~/$ProjectDirectory/$FastqDirectory/realigned_reads.bam
    -knownSites ~/$ProjectDirectory/$FastqDirectory/filtered_snps.vcf
    -knownSites ~/$ProjectDirectory/$FastqDirectory/filtered_indels.vcf
    -BQSR ~/$ProjectDirectory/$FastqDirectory/recal_data.table
    -o ~/$ProjectDirectory/$FastqDirectory/post_recal_data.table
```

8.7) Etapa 14: Analise de Covariantes (Analyze Covariates)

```
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar --maximum_cycle_value 35000
    -T AnalyzeCovariates -R $REFERENCE
    -before ~/$ProjectDirectory/$FastqDirectory/recal_data.table
    -after ~/$ProjectDirectory/$FastqDirectory/post_recal_data.table
    -plots ~/$ProjectDirectory/$FastqDirectory/recalibration_plots.pdf
```

8.8) Etapa 15: Aplicar BQSR

```
    java -jar ~/programs/gatk/GenomeAnalysisTK.jar -T PrintReads
    -R $REFERENCE -I ~/$ProjectDirectory/$FastqDirectory/realigned_reads.bam
```

```
-BQSR ~/$ProjectDirectory/$FastqDirectory/recal_data.table
-o ~/$ProjectDirectory/$FastqDirectory/recal_reads.bam
```

8.9) Etapa 16: Identificar Variantes (Call Variantes)

```
java -jar ~/programs/gatk/GenomeAnalysisTK.jar
-T HaplotypeCaller -R $REFERENCE
-I ~/$ProjectDirectory/$FastqDirectory/recal_reads.bam
-o ~/$ProjectDirectory/$FastqDirectory/raw_variants_recal.vcf
```

8.10) Etapa 17: Extrair SNPs e Indels

8.10.1) SNPs

```
java -jar ~/programs/gatk/GenomeAnalysisTK.jar
-T SelectVariants -R $REFERENCE
-V ~/$ProjectDirectory/$FastqDirectory/raw_variants_recal.vcf
-selectType SNP -o ~/$ProjectDirectory/$FastqDirectory/raw_snps_recal.vcf
```

8.10.2) indels

```
java -jar ~/programs/gatk/GenomeAnalysisTK.jar -T SelectVariants
-R $REFERENCE -V ~/$ProjectDirectory/$FastqDirectory/raw_variants_recal.vcf
-selectType INDEL -o ~/$ProjectDirectory/$FastqDirectory/raw_indels_recal.vcf
```

8.11) Etapas 18 e 19: Filtrar SNPs e indels

8.11.1) SNPs

```
java -jar ~/programs/gatk/GenomeAnalysisTK.jar -T VariantFiltration -R $REFERENCE
-V ~/$ProjectDirectory/$FastqDirectory/raw_snps_recal.vcf
--filterExpression
'QD < 2.0 || FS > 60.0 || MQ < 40.0 || MQRankSum < -12.5 ||
ReadPosRankSum < -8.0 || SOR > 4.0'
--filterName "basic_snp_filter"
-o ~/$ProjectDirectory/$FastqDirectory/filtered_snps_final.vcf
```

8.11.2) Indels

```
java -jar ~/programs/gatk/GenomeAnalysisTK.jar
-T VariantFiltration -R $REFERENCE
-V ~/$ProjectDirectory/$FastqDirectory/raw_indels_recal.vcf
--filterExpression 'QD < 2.0 || FS > 200.0 || ReadPosRankSum < -20.0 || SOR > 10.0'
--filterName "basic_indel_filter"
-o ~/$ProjectDirectory/$FastqDirectory/filtered_indels_recal.vcf
```

9) Etapa 20: Anotacao de SNPs e Predicao de Efeitos Biologicos

```
    java -jar snpEff.jar
    -v snpeff_db ~/$ProjectDirectory/$FastqDirectory/filtered_snps_final.vcf >
    ~/$ProjectDirectory/$FastqDirectory/filtered_snps_final.ann.vcf
```

10) Etapa 21: Computar Estatistica de Cobertura de Medias

```
    bedtools genomecov -bga -ibam ~/$ProjectDirectory/$FastqDirectory/recal_reads.bam >
    ~/$ProjectDirectory/$FastqDirectory/genomecov.bedgraph
```

11) Etapa 22: Compilar Estatistica

```
  ## Fim da "for loop"
done
```