Evaluación de diferentes herramientas de detección de variantes puntuales (SNV e INDELS) sobre datos de secuenciación masiva procedentes de exoma.

Eduardo Candeal Núñez
Máster Bioinformática y Bioestadística
Área 1

Nombre Consultor/a: Joan Maynou Fernández
Nombre Profesor/a responsable de la asignatura: Javier Luís Cánovas Izquierdo

04/06/2019

Licencias alternativas (elegir alguna de las siguientes y sustituir la de la página anterior)

A) Creative Commons:

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-NoComercial 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-SinObraDerivada 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento-CompartirIgual 3.0 España de Creative Commons

Esta obra está sujeta a una licencia de Reconocimiento 3.0 España de Creative Commons

B) GNU Free Documentation License (GNU FDL)

Copyright © AÑO TU-NOMBRE.

Permission is granted to copy, distribute and/or modify this document under the terms of the GNU Free Documentation License, Version 1.3 or any later version published by the Free

C) Copyright

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Evaluación de diferentes herramientas de detección de variantes puntuales (SNV e INDELS) sobre datos de secuenciación masiva procedentes de exoma.*
Nombre del autor:	*Eduardo Candeal Nuñez*
Nombre del consultor/a:	*Joan Maynou Fernández*
Nombre del PRA:	Javier Luís Cánovas Izquierdo
Fecha de entrega (mm/aaaa):	04/06/2019
Titulación::	*Bioinformática y bioestadística*
Área del Trabajo Final:	*Área 1*
Idioma del trabajo:	*Castellano*
Palabras clave	*- Next Generation Sequencing (NGS)* *- Germline Mutation* *- Benchmarking Variant-Calling*

Resumen del Trabajo (máximo 250 palabras): *Con la finalidad, contexto de aplicación, metodología, resultados i conclusiones del trabajo.*

El presente trabajo trata de la evaluación de diferentes herramientas para la detección de variantes, detecciones puntuales de bases (SNV) e inserciones y deleciones (INDEL) en datos procedentes de secuenciación de Exoma de *Homo sapiens,* utilizando la plataforma de Illumina HiSeq. Para el desarrollo del trabajo se ha contado con la muestra control *NA12878 procedente del proyecto "Genome in a bottle*" usada como estándar en este tipo de estudios, su genotipo estándar y el genoma de referencia GRCH37.

Se evaluaron un total de 5 detectores (Freebayes, Samtools, GATK-HC, SNVer y VarScan) siguiendo el mismo tratamiento en cada uno de los detectores, es decir, aplicando el mismo pipeline.

Los resultados obtenidos mostraron que los mejores detectores clasificados por sensibilidad fueron por este orden, Samtools, FreeBayes y GATK-HC, mientras que la clasificación de los mejores por especificidad fueron FreeBayes, VarScan y Samtools.

Como tarea final se realizó la fusión de los distintos ficheros que contenían variantes (VCF) para mejorar los resultados previos. El análisis de esta tarea mostró un incremento en la especificidad con un valor de 0,9723, aunque no llegó a mejorar los resultados de sensibilidad respecto al mejor detector, Samtools, con un valor de 0,7623 frente a 0,7735.

Por lo tanto, la conclusión que se puede establecer es que para obtener una mejor sensibilidad, se optaría por el detector de Samtools, mientras que para obtener mejores valores de especificidad seleccionariamos la fusión de los tres mejores detectores (Samtools, Freebayes y GATK-HC) obtenida por medio de la unión de sus distintos ficheros.

Abstract (in English, 250 words or less):

The present work is based on the evaluation of different tools for the variant calling detection, Single Nucleotide Variant (SNV) and insertions and deletions (INDEL) in the data of Exome sequencing from *Homo sapiens* with the platform of Illumina HiSeq. For this purpose, the control sample NA12878 from "Genome project in a bottle" was used which is used as standard in this type of studies, as well as its standard genotype, and finally the GRCH37 as the reference genome.

A total of 5 detectors (Freebayes, Samtools, GATK-HC, SNVer and VarScan) were evaluated following the same variant calling pipeline for each one of them, that is, applying the same pipeline.

The results indicate that the most sensible detectors were, in order, Samtools, FreeBayes and GATK-HC, while the classification of the most specific were FreeBayes, VarScan and Samtools.

As final task, it was intended to merge the different files that contained the variants (VCF) from each detector, in order to improve the previous results. The results showed the specificity value increased up to 0.9723, however this tend was not replicate for sensitivity results, as the value for best detector, Samtools, dropped to 0.7623 from 0.7735.

Taking into account this results, it can be established the conclusion that in case of being interested on obtaining better sensitivity for variant calling detection, Samtools detector should be chosen, while to obtain better specificity values the merged file from best detectors (Samtools, Freebayes and GATK-HC) should be selected.

Índice

Lista de figuras

1. Introducción

1.1 Contexto y justificación del Trabajo

El contexto de este trabajo fin de master se encuentra dentro del campo de análisis de secuencias de alto rendimiento a partir de las nuevas tecnologías de secuenciación masiva, conocidas por sus siglas en ingles NGS, Next Generation Sequencing.

La principal ventaja que ofrece le tecnología NGS frente a la clásica secuenciación Sanger es el coste por de secuenciación por megabase. Mientras que el proyecto genoma humano costo entre 1-3 mil millones de dólares durante un periodo de 15 años, se estima que el coste del primer genoma de un individuo se elevo hasta los 80 millones de dólares. Ahora con las nuevas tecnologías de secuenciación masiva podemos obtener un genoma completo por menos de 2000 dólares como se puede apreciar en la Imagen 1.

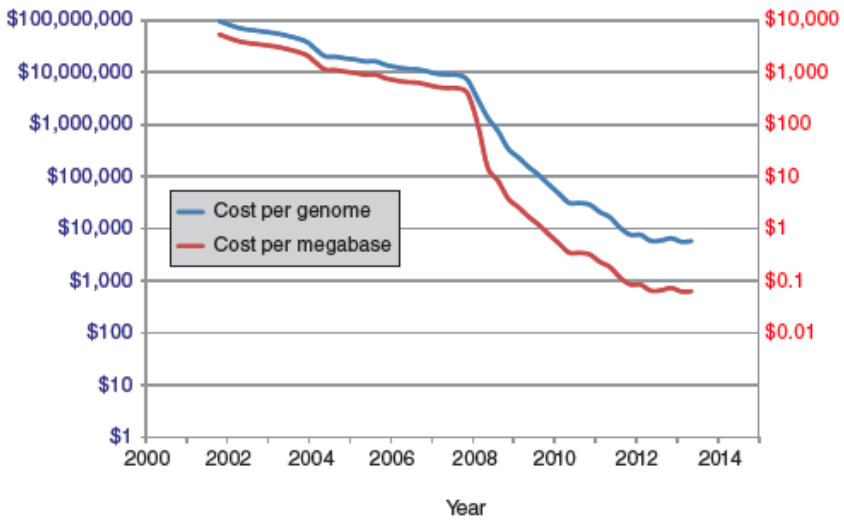


Imagen 1- Comparación de coste por megabase (10^6) y por genoma completo a lo largo de los últimos años. [6]

Existen diferentes plataformas de secuenciación masiva que podemos clasificar en función del tamaño de lectura generado:

-Tamaño pequeño (50-150 pb)
- Illumina (150 pb).
- Secuenciación por ligación(SOLID) (50pb).

-Tamaño grande (700 pb)
- Pirosecuenciación 454 Life Sciences.

-Tamaño muy grande
- Pacific Biosciences entre 10.000 -60.000 pb.

La tecnología NGS se puede utilizar tanto secuenciación dirigida, es decir, de genes en concreto (paneles NGS), secuenciación del exoma completo, WES,

(parte codificante del genoma) o la secuenciación del genoma completo (WGS).

En el presente trabajo nos vamos a focalizar en la plataforma de secuenciación de Illumina HiSeq y se analizaran las secuencias procedentes del exoma completo (WES).

Se ha visto que con las nuevas tecnologías de secuenciación hemos podido reducir los costes económicos y en tiempo, sin embargo el nuevo reto que se plantea con estas tecnologías está en el análisis bioinformático, ya que es indispensable en la detección correcta de las variantes de una base (SNV) o inserciones o delecciones de bases (INDELS).

Los datos procedentes de NGS tiene que ser procesados para poder obtener un output evaluable. Para ello es necesario definir y aplicar pipeline de análisis a los datos de NGS. Este trabajo es esencial a la hora de obtener los resultados correctos y reproducibles.

Como se puede apreciar en la Imagen 2, el análisis bioinformatico se situa dentro del workflow de un experimento de secuenciación masiva. Diferentes pipelines pueden desembocar en que los resultados en la detección de variantes sean distintos. Por ello la correcta elección de nuestra herramientas bioinformáticas para el tipo de análisis que queramos llevar es fundamental.
Para la realización del TFM creemos adecuado utilizar como referencia las buenas prácticas definidas en el Broad Institute (https://software.broadinstitute.org/gatk/best-practices/).

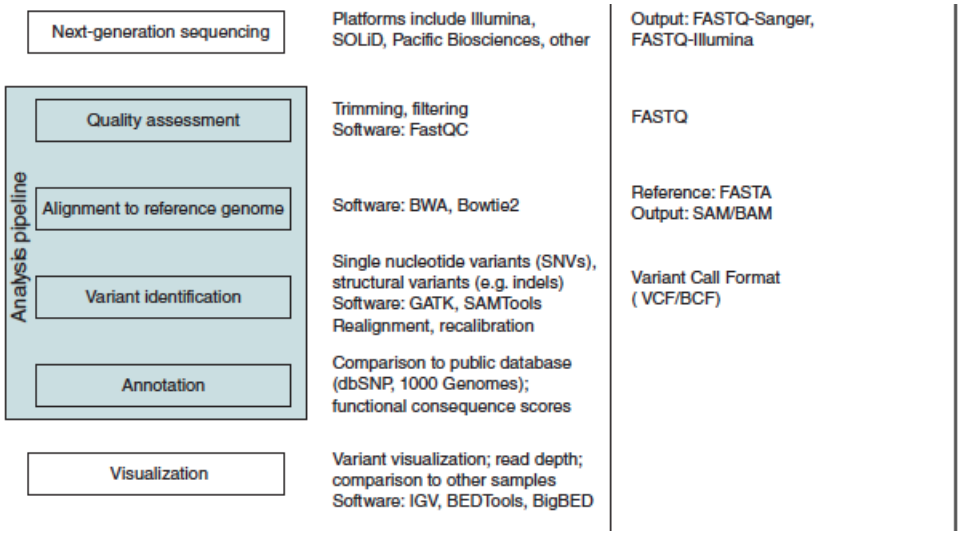


Imagen 2- Workflow del análisis bioinformático de datos procedentes de secuenciación masiva [6]

Durante el TFM nos vamos a centrar en las etapas correspondientes al análisis bioinformático, es decir comenzaría en la primer etapa de obtención de fichero FastQ que contiene la lecturas procedentes de la secuenciación de tamaño variable en función de la preparación de librerías. Cada lectura de base contiene la calidad de lectura de esta. Esto nos permite realizar el control de

calidad de estas lecturas y filtrar por calidad de base. También hay que considerar que necesitaremos eliminar las primeras bases de los reads que contiene los adaptadores.

La segunda etapa es el de alineamiento frente al genoma de referencia, por la cual obtendríamos un fichero SAM/BAM que contiene el alineamiento de las lecturas frente al genoma de referencia
Posterior a ello se realizará un postprocesado donde se eliminarán duplicados y se seleccionaran las zonas de interés.

La tercera etapa consiste en la obtención por el procesamiento de nuestro fichero SAM/BAM de la variaciones puntuales o pequeñas inserciones o delecciones que contiene la muestra a analizar, que desemboca en un fichero que contiene estas variaciones llamado Variant Calling Format (VCF).

Finalmente este fichero que contiene las variantes puede ser procesados con otras herramientas bioinformáticas para obtener resultados estadísticos.

1.2 Objetivos del Trabajo

2.1. Objetivos Generales

Evaluación del rendimiento de los diferentes detectores usados para las búsqueda de SNV e INDELs de datos procedentes de secuenciación masiva de exoma.

2.2. Objetivos específicos

1.0- Descripción y búsqueda en la bibliografía de la tecnología de secuenciación masiva y las técnicas bioinformáticas involucradas en el procesamiento de la información.

1.1-*Decargar las herramientas/software bioinformáticos necesarios para las tareas propuestas. Descargar el genoma de referencia y muestra usada como control, NA12878. Testear cada una de las herramientas descargadas.*

1.2- Obtención de resultados de alineamiento (fichero SAM/BAM) y procesamiento de este eliminando duplicados.

1.3- Creación de los distintos ficheros VCF con las diferentes herramientas de detección de variantes propuestas.

1.4-Obtención de resultados estadísticos a partir de la herramienta RTG Command Reference y comparación de cada fichero VCF respecto VCF gold estándar y clasificación en función de sensibilidad y especificidad.

1.5- Fusión de todos los ficheros VCF obtenido por los distintos detectores para la obtención de un fichero global.

1.6-Obtención de los resultados estadísticos a través de la herramienta RTG Command Reference y comparación del VCF fusionado frente al fichero VCF gold estándar y clasficiación en función de sensibilidad y especificidad.

1.3 Enfoque y método seguido

Para la realización de este TFM se propone utilizar el siguiente workflow.

Estrategia o Workflow.

1- Utilizar un alineador como estándar, BWA-MEM[5], para la etapa de alineamiento de las lecturas.

2- Procesamiento del fichero SAM con 5 detectores de variantes seleccionados para obtener el fichero VCF.

3-Comparación del VCF frente al VCF del genotipo estándar.

4- Evaluación y clasificación de cada uno de los detectores utilizados.

5-Fusión de todos los VCF de los distintos detectores.

6- Evaluación y clasificación de cada uno del VCF global.

Se ha decido utilizar esta estrategia porque seguiremos las buenas prácticas del broad institute para el análisis del genoma (gatk). Por eso vamos a usar el alineador Gold estándar BWA-MEM recomendado en estas mejores practicas de workflow.

Se ha optado por hacer una evaluación final del conjunto de todos los detectores, fusionando todos los ficheros VCF obtenidos y evaluar si se consigue mejor rendimiento que cuando se evalúa cada uno de ellos por separado.

1.4 Planificación del Trabajo

4.1. Tareas

Objetivo 1.1

1. Descargar de las distintas herramientas bioinformáticas propuestas. BWA-MEM, Free-Bayes, VarScan, GATK-HC, Deep Variant, SAMtools y RTG Command Reference.

2. Descargar de la muestra FastQ NA12878 de la plataforma de secuenciación Illumina HiSeq del repositorio Genome in a Bottle usando la plataforma de secuenciación Ilummina.
3. Instalación de software BWA-MEM, en el equipo local.
4. Instalación de las herramientas necesarias para la detección de variantes. FreeBayes, VarScan, GATK-HC, Deep Variant, SAMtools y RTG Command Reference.
5. Testear el funcionamiento de cada herramienta tras instalación.
6. Descargar del genoma de referencia. GRCh37, en formato Fasta.

Objetivo 1.2

7. Uso de la herramienta BWA-MEM para crear el índice a partir del genoma de referencia.
8. Creación del índice a partir del genoma de referencia GRCh37 usando el alineador estándar BWA-MEM.
9. Alineamiento de la muestra NA12878 con BWA-MEM frente índice creado a partir del genoma de referencia. Obtención del fichero SAM/BAM.
10. Pre-procesado del BAM. Eliminación de duplicados y selección de zonas de interés. Uso de las Picard tools.

Objetivo 1.3

11. Procesamiento del fichero SAM obtenido con las diferentes herramientas de detección de variantes (FreeBayes, VarScan, GATK-HC, Deep Variant y SAMtools) para la obtención de los distintos fichero VCF.
12. Estudio del formato VCF. Detallar información relevante para la detección de SNV e INDELS.

Objetivo 1.4

13. Procesamiento de los ficheros VCF a través de la herramienta RTG Command Reference.
14. Obtención de resultados usando la opción vcfstats de la herramienta RTG.
15. Comparación de cada fichero VCF respecto al fichero VCF procedente del Gold estándar.
16. Evaluación y clasificación de los resultados de SNV e INDELs obtenidos tras la comparación. Cuantificando los verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos.

Objetivo 1.5

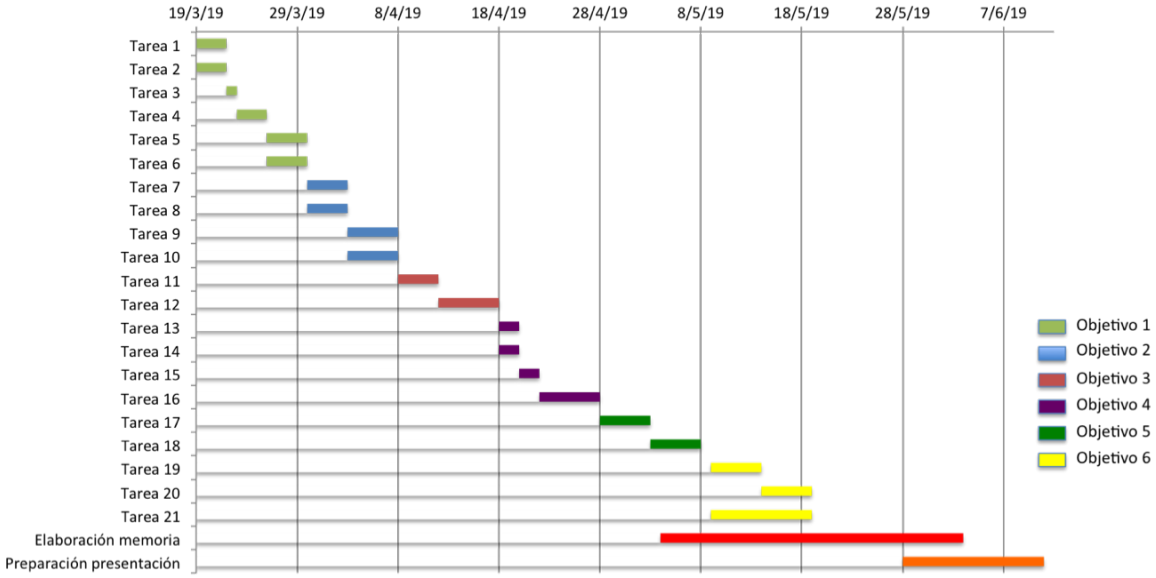
17. Fusión de los distintos ficheros VCF para la obtención de un VCF global de todas las herramientas utilizadas.
18. Comparación del fichero VCF global respecto al fichero VCF procedente del Gold estándar.

Objetivo 1.6

19. Evaluación y clasificación de los resultados de SNV e INDELs obtenidos tras la comparación entre el VCF global y VCF Gold estándar. Cuantificando los verdaderos positivos, falsos positivos, verdaderos negativos y falsos negativos.
20. Clasificación final de todos los resultados de SNV e INDELs.
21. Establecimiento de conclusiones en función de los diferentes resultados.

4.2. Calendario:

Diagrama de Gantt



4.3. Hitos:

Se plantearon 4 Hitos:

- Obtención del fichero SAM/BAM procesado que contiene los alineamientos de las lecturas frente al genoma de referencia.

- Generación de los ficheros VCF de cada detector de variantes utilizados, cinco en total.

- Obtención de resultados de sensibilidad y especificidad de cada detector en comparación con el genotipo estándar.

- Generación de un VCF global a través de la fusión de todos los ficheros VCF de cada uno de los detectores usados y obtención de resultados de sensibilidad y especificad globales.

1.5 Breve sumario de productos obtenidos

- Alineamiento del genoma de referencia Homo_sapiens.GRCh37 con la muestra NA12878 mediante el mapeador BWA-MEM. Obtención del BAM.

- Procesado del BAM crudo por calidad de base, eliminación de duplicados y selección de zonas de captura de los exones.

- Generación de los distintos ficheros VCF obtenidos a partir de del usos de las distintas herramientas para la detección de variantes.

- Comparación frente al Genotipo estándar de SNP e INDELs de cada uno de los detectores de variantes usados. Obtención de resultados de Falsos positivos, Falsos negativos, Sensibilidad y Especificidad.

- Fusión de todos los detectores para la obtención del VCF global a través de la herramienta CombineVariants de GATK tools (versión 3,8).

- Comparación con el Genotipo estándar del VCF global para la intersección de los 5 detectores y evaluación de resultados.

- Separación de resultados entre SNP e INDELS de cada uno de los detectores para la evaluación frente al genotipo estándar.

- Generación de un nuevo VCF con los tres mejores detectores para SNP e INDELS.

- Comparación con el Genotipo estándar del VCF global para la intersección de los 3 mejores detectores y evaluación de resultados.

1.6 Breve descripción de los otros capítulos de la memoria

El resto de capítulos se componen de las etapas necesarias durante todo el TFM para la realización de este mismo y la descripción de los resultados obtenidos. Se ha dividido en un total de 9 secciones, que son:

1-Descarga y obtención de cada una de las herramientas/software necesarias para la realización del trabajo así como el genoma de referencia y la muestra usada como control, NA12878.

2-Control de Calidad de los datos crudos (FASTQ) de la muestra usada como referencia.

3-Alineamiento de la muestra frente al Genoma de referencia.

4-Preprocesado del fichero de alineamiento BAM crudo.

5- Uso de las diferentes herramientas para la detección de variantes.

6-Obtención de resultados estadísticos.

7-Comparación frente al genotipo estándar.

8-Generación del VCF global a partir de los VCF de los 5 detectores.

9-Evaluación del VCF global.

2. Resto de capítulos

A continuación se describe el proceso por el cual se han ido obteniendo los distintos resultados de este trabajo fin de Máster.

Debido a que uno de los objetivos de esta trabajo era establecer un workflow que describa cada uno de los pasos obtenido se va a mostrar en detalle todos los pasos obtenidos hasta la consecución de los resultados finales.

1-Descarga y obtención de cada una de las herramientas/software necesarias para la realización del trabajo así como el genoma de referencia y la muestra usada como control, NA12878.

Para la realización de este TFM fue necesario la descarga de las siguientes herramientas.
-FreeBayes [18].
-GATK-HC [20].
-Samtools/bcftools[11, 12].
-VarScan [19].
-SNVer (anteriormente fue propuesta otra herramienta DeepVariant) [17].

A continuación se muestran los comandos usados para la descarga y testeo de cada herramienta:

-FreeBayes.

$git clone --recursive git://github.com/ekg/freebayes.git
$cd freebayes
$make

Comprobar que el software ha sido correctamente instalado, ejecutando el siguiente comando:
$bin/freebayes

-GATK-HC

Descargamos las gatk tools y ejecutamos el siguiente comando.
`$java -jar gatk-package-4.1.0.0-local.jar HaplotypeCaller`

-Samtools

Descargamos las samtools y ejecutamos los siguientes comandos.

$cd samtools-1.6

$./configure --prefix=/where/to/install
$make
$make install

-VarScan

Decargamos la ultima versión y ejecutamos el siguiente comando
$ java -jar VarScan.v2.3.9.jar

-SNVever

Descarga del 5º detector SNVer del siguiente directorio: http://snver.sourceforge.net/
Para el uso de este detector se realiza la invocación del siguiente comando:
$java -jar SNVerIndividual.jar

A parte de las herramientas para la detección de variantes fueron descargados otras herramientas como:

-BWA. Software necesario para el alienamiento de los reads [5].
-FastQC. Herramienta para analizar de los datos FastQ de origen [13].
-IGV. Herramienta necesaria para la visualización de los alineamientos [9,10].
-Cutadapt. Herramienta necesaria para el tratamiento de las muestras FastQ [8].

Finalmente fue necesario la descarga del genoma de referencia y de la muestra usada como control como se describe a continuación.

Descarga de la muestra como control NA12878:

Para ello descargamos del proyecto genome in a bottle,es un consorcio público-privado-académico organizado por el NIST para desarrollar la infraestructura técnica que permite la traducción de toda la secuenciación del genoma humano a la práctica clínica.

Dentro del enlace https://github.com/genome-in-a-bottle, accedemos al directorio giab_data_indexes/NA12878/**sequence.index.NA12878_Illumina_HiSeq_Exome_Garvan_fastq_09252015** donde encontramos las muestras de Exoma de la muestra NA12878.

Descargamos los dos ficheros correspondientes del siguiente ftp:

ftp://ftp-trace.ncbi.nih.gov/giab/ftp/data/NA12878/Garvan_NA12878_HG001_HiSeq_Exome/NIST7035_TAAGGCGA_L001_R1_001.fastq.gz

ftp://ftp-trace.ncbi.nih.gov/giab/ftp/data/NA12878/Garvan_NA12878_HG001_HiSeq_Exome/NIST7035_TAAGGCGA_L001_R2_001.fastq.gz

Descarga del genoma de referencia:

Descarga del genoma de referencia en el siguiente ftp, ftp://ftp.ensembl.org/pub/grch37/current/fasta/homo_sapiens/cds/

2-Control de Calidad de los datos crudos (FASTQ) de la muestra usada como referencia.

Previamente a iniciar el alineamiento frente al genoma de referencia, necesitamos realizar un control de calidad de nuestras muestras fastq [Imagen 3]. A partir de la herramienta FastQC podemos analizar nuestros datos crudos de origen. Una vez visto como están los datos de origen se procede a realizar un tratamiento de las muestras que consiste en:

-1) Eliminación de los adaptadores:

-2) Eliminación de las primeras bases de mala calidad:

-3) Eliminación de bases de baja calidad.

Para ello se usa la herramienta cutadapt [8] , que nos permite realizar todo estos procesos.

Control de calidad de los fastq previo al pre-procesado:

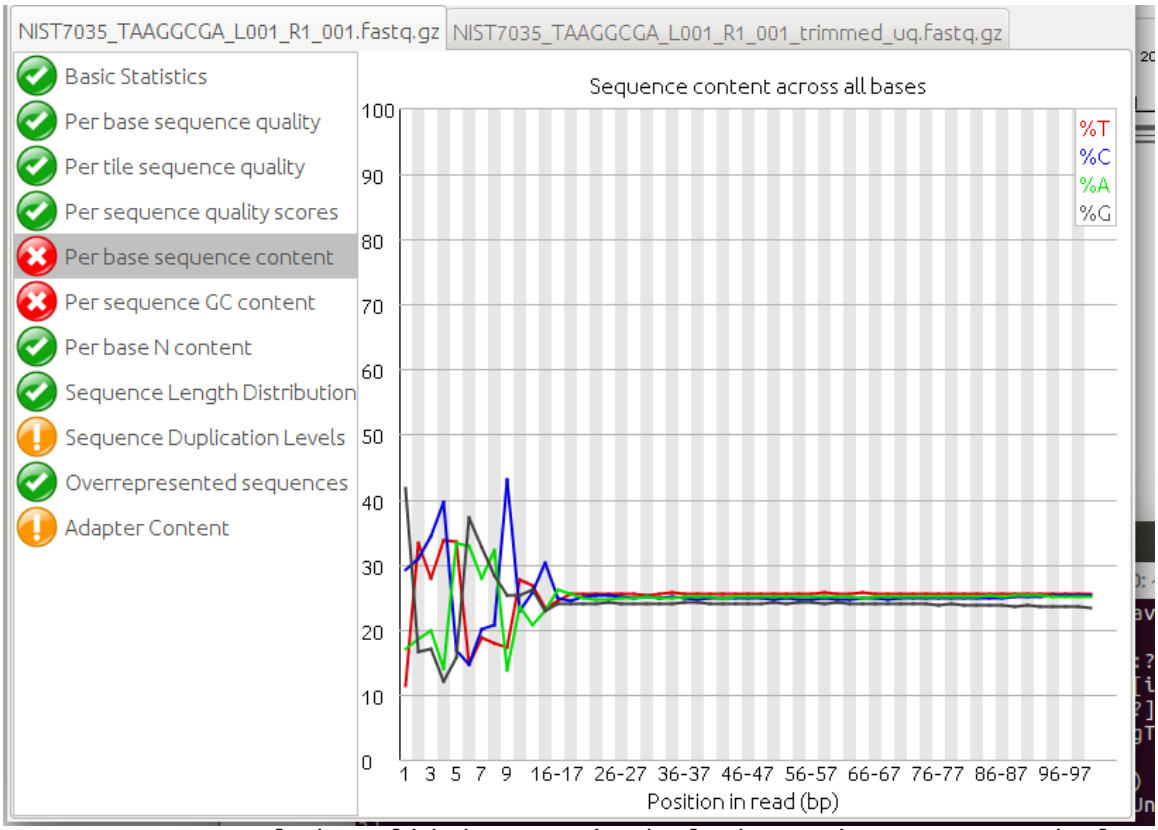


Imagen 3. Control de Calidad a través de la herramienta FastQC de los muestras fastq previa al preprocesado.

1) ./cutadapt -a CTGTCTCTTATACACATCTCCGAGCCCACGAGAC -A CTGTCTCTTATACACATCTGACGCTGCCGACGA -o NIST7035_TAAGGCGA_L001_R1_001.fastq_trimmed.gz -p NIST7035_TAAGGCGA_L001_R2_001_trimmed.fastq.gz NIST7035_TAAGGCGA_L001_R1_001.fastq.gz NIST7035_TAAGGCGA_L001_R2_001.fastq.gz

Una vez obtenido los fastq con la eliminación de los adaptadores usados por Nextera, procedemos al eliminación de las primeras bases y reads de baja calidad. Estos nos va a generar reads desparejados que obtendremos mediante las opciones --too-short-output –too-short-paired-output, como se muestra.

2-3)./sudo cutadapt -u 15 -U 15 -q 10 -m 50 -o NIST7035_TAAGGCGA_L001_R1_001_trimmed_uq.fastq.gz -p NIST7035_TAAGGCGA_L001_R2_001_trimmed.fastq_uq.gz --too-short-output NIST7035_TAAGGCGA_L001_R1_001_trimmed_se.fastq.gz --too-short-paired-output NIST7035_TAAGGCGA_L001_R2_001_trimmed_se.fastq.gz NIST7035_TAAGGCGA_L001_R1_001_trimmed.fastq.gz NIST7035_TAAGGCGA_L001_R2_001_trimmed.fastq.gz

Los parámetros aplicados fueron eliminación de las 15 primeras bases, filtro por 10 de calidad fijando un tamaño de read de 50.

Control de calidad <u>después de realizar las tareas de procesado</u> de los fastq [Imagen 4].

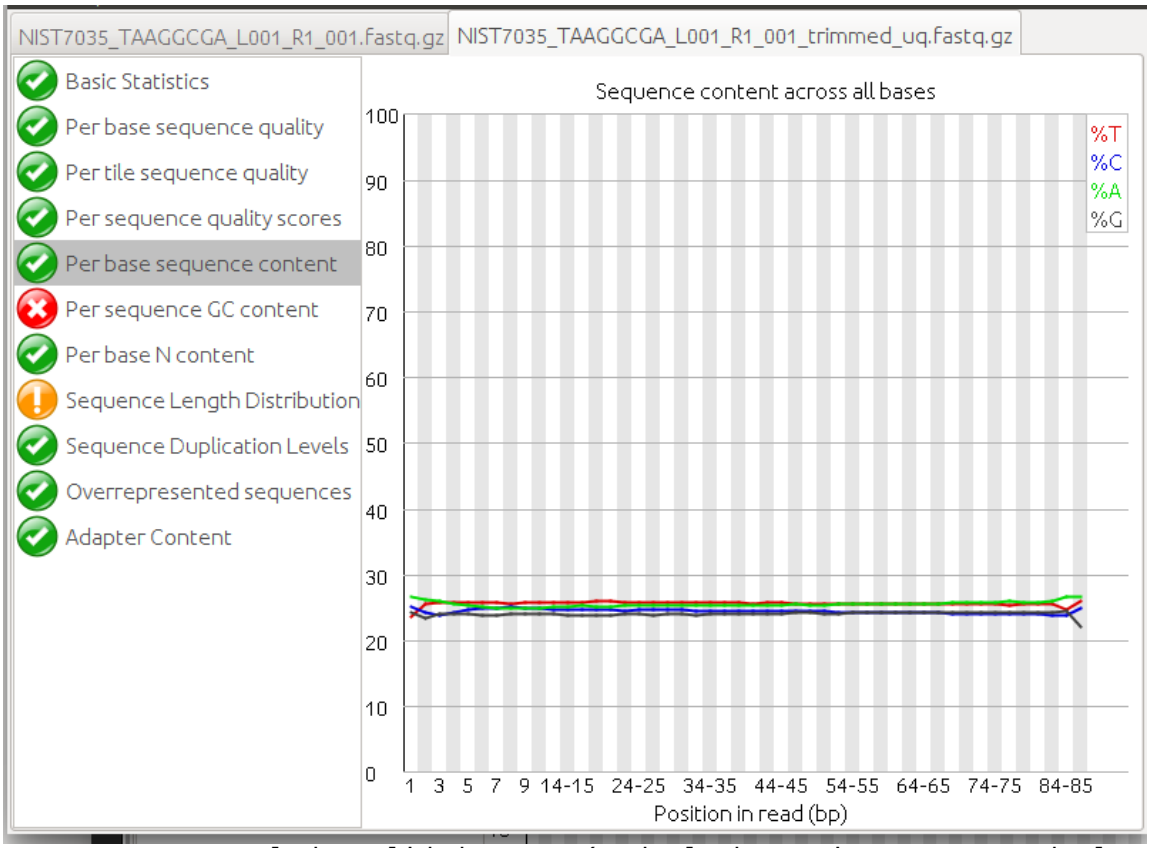


Imagen 4. Control de Calidad a través de la herramienta FastQC de los muestras fastq después del tratamiento de preprocesamiento.

3-Alineamiento de la muestra frente al Genoma de referencia.

<u>3.1 Crear indice:</u>

Primero antes de alinear frente al genoma de refencia necesitamos crear el indice con el mapeador BWA pasándole nuestro genoma de referencia [Imagen 5].

./bwa index -a bwtsw 'Homo_sapiens.GRCh37.dna.primary_assembly.fa.gz'

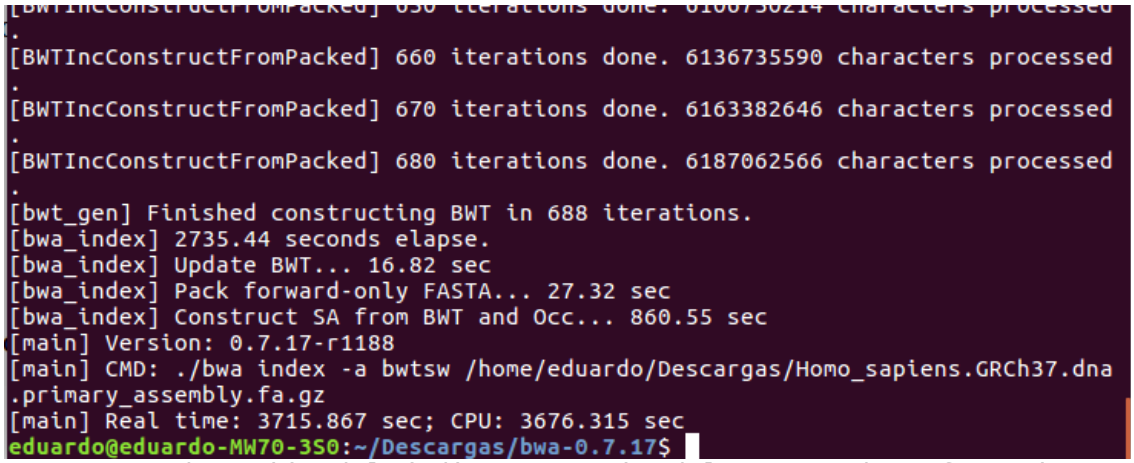


Imagen 5. Obtención del índice a partir del genoma de referencia por el mapeador BWA mediante la opción index.

3.2. Alineamientos mediante BWA:

Debido a que hemos obtenido distintos fastq, los paired-end y los single-end, necesitamos realizar 3 alineamientos distintos (paired end juntos y los dos single-end por separado).

Como queremos obtener directamente el bam ordenado con output aplicamos un pipeline a través de las samtools.

Obtención directa del BAM sort

Ejemplo de single-end:

$ ./bwa mem -M -t 4 /Homo_sapiens.GRCh37.dna.primary_assembly.fa.gz NIST7035_TAAGGCGA_L001_R1_001_trimmed_se.fastq.gz | samtools sort -O BAM -o output_se1.bam

Ejemplo paired-end:

$./bwa mem -M -t 4 Homo_sapiens.GRCh37.dna.primary_assembly.fa.gz NIST7035_TAAGGCGA_L001_R1_001_trimmed_uq.fastq.gz NIST7035_TAAGGCGA_L001_R2_001_trimmed.fastq_uq.gz| samtools sort -O BAM -o output_pe.bam

Una vez obtenido los tres BAM procedemos a la fusión de estos a través de la opción merge de las samtools

$ samtools merge output_merge.bam output_se1.bam output_se2.bam output_pe.bam

4-Preprocesado del fichero de alineamiento BAM crudo.

Filtro del BAM reads por calidad mínima de base (30):

$ samtools view -bq 30 output_merge.bam >output_Q30.bam

Usamos la opción b para obtener un bam y la opción de q para establecer el valor numérico por el cual vamos a filtrar los reads por calidad.

Visualizar con IGV(Imagen 6)

Para ello previamente tenemos que generar el índice mediante la herramienta samtools.

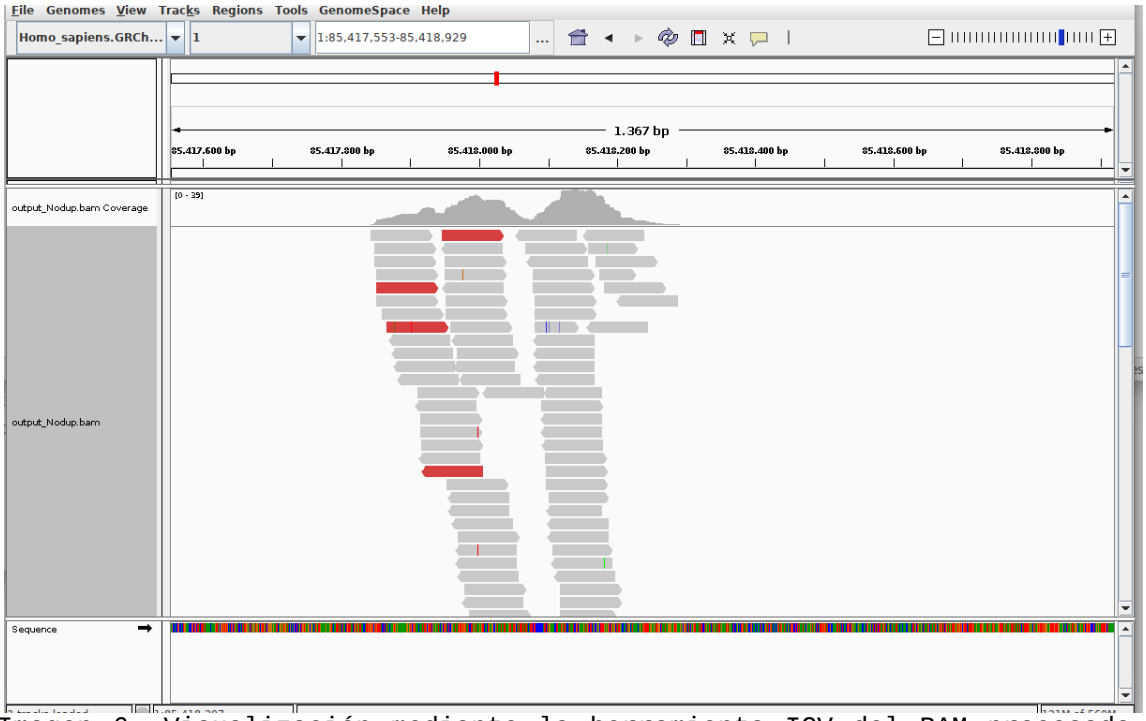


Imagen 6. Visualización mediante la herramienta IGV del BAM procesado frente al genoma de referencia en un zona exónica.

Selección de las zonas de captura:

Para ello se descargo la herramienta bedtools [14], se instalo y se ejecuto el siguiente comando.

$/intersectBed -abam 'output_Q30.bam' -b 'highconf_nextera_intersection_test.bed'>'outputbed.bam'

Eliminación de reads duplicados:

Para ello se descargaron las picardtools [16] y se ejecuto el siguiente comando:

$java -jar picard.jar MarkDuplicates INPUT='outputbed.bam' OUTPUT='outputbed_Nodup.bam' METRICS_FILE=metrics.txt

Esto genero un output de 1,7 GB, de mayor tamaño al original. Por lo tanto se probo a realizar este paso a través de las samtools mediante la opción rmdup y -S (trata igual los paired end que los single end).

$samtools rmdup -S outputbed.bam output_Nodup.bam

5- Uso de las diferentes herramientas para la detección de variantes.

Después de procesar el BAM, eliminando reads duplicados, seleccionando las zonas de captura y seleccionando los reads por calidad mínima de base se procedió a pasarlas por cada una de las herramientas para la detección de variantes.

1) Freebayes

$ ./freebayes -f 'Homo_sapiens.GRCh37.dna.chromosome.1.fa' 'outputbed_Nodup.bam' >'freebayes.vcf'

2) VarScan:

Para procesar con esta herramienta previamente hay que realizar con las samtools un fichero intermedio llamado mpileup, a partir del bam y la referencia de origen.

$samtools mpileup -f 'Homo_sapiens.GRCh37.dna.chromosome.1.fa' 'outputbed_Nodup.bam' >'mpileup'

Posteriormente, una vez obtenido el fichero mpileup se obtuvo por separado los SNP e INDELS. Se ejecuto de nuevo a partir de la herramienta VarScan los siguientes comandos:

Generación los SNPs

$java -jar VarScan.v2.3.9.jar mpileup2snp muestra.mpileup —min-reads2 0 —min-avg-qual 20 —min-coverage 10 —min-var-freq 0.10 —p-value 0.10 --output-vcf 1>varscan_snp.vcf

Posteriormente generó los INDELS

$java -jar VarScan.v2.3.9.jar muestra.mpileup —min-reads2 0 —min-avg-qual 20 —min-coverage 10 —min-var-freq 0.10 —p-value 0.10 --output-vcf 1>varscan_indel.vcf

3) GATK-HC:

Al ejecutar directamente la herramienta con el último BAM obtenido a partir de la eliminación de duplicados, dio el siguiente error:

“java.lang.IllegalArgumentException: samples cannot be empty”

Buscando se encontró que previamente es necesario hacer un pretratamiento del fichero BAM con las picard tools usando la opción AddOrReplaceReadGroups. Esto sirve para agrupar todos los reads en un grupo [15].

$ java -jar picard.jar AddOrReplaceReadGroups I= 'output_Nodup.bam' O= 'output_picard.bam' RGID=4 RGLB=lib1 RGPL=illumina RGPU=unit1 RGSM=20

Posteriormente se obtuvo el BAM para ser procesado mediante la opción HaplotypeCaller, para ello también fue necesario crear el index.

$samtools index output_picard.bam

$java -jar gatk-package-4.1.0.0-local.jar HaplotypeCaller -R Homo_sapiens.GRCh37.dna.chromosome.1.fa -I output_picard.bam -O gatk.vcf

4) Samtools/bedtools:

Necesitamos utilizar el fichero “mpileup” para procesar a través de las samtools.

$samtools mpileup -uf Homo_sapiens.GRCh37.dna.chromosome.1.fa output_Nodup.bam | bcftools call -mv > samtools.vcf

5) SNvever:

Para el uso de este detector se realiza la invocación del siguiente comando:

$java -jar SNVerIndividual.jar -i output_Nodup.bam –r Homo_sapiens.GRCh37.dna.chromosome.1.fa -o /SNVever.vcf

Ello nos genera distintos ficheros donde está los SNV y los INDELS, como en el caso del VarScan generamos la fusión de los dos usando de nuevo la herramienta RTG-Command.

Filtrado de variantes en los ficheros VCF:

A partir de la herramienta RTG command eliminamos las variantes que no superen ciertos valores de calidad (Profundidad media de 8 y Mínima calidad de base de 20).

Ejemplo para el detector Freebayes:

$java -jar RTG.jar vcffilter -i 'freebayes.vcf' -o freebayes.vcf.gz -q 20 -d 8

En el caso del detector VarScan y Snever hubo que realizar pasos extra ya que nos genero por separado SNPs e INDELs.

Caso VarScan y SNever:

Se realiza el filtrado de ambos vcf como se hizo en los demás ficheros

$java -jar RTG.jar vcffilter -i varscan_indel.vcf -o varscan_indel_filter.vcf.gz -q 20 -d 8

$java -jar RTG.jar vcffilter -i varscan_snp.vcf -o varscan_snp_filter.vcf.gz -q 20 -d 8

Finalmente se realiza la fusión de ambos ficheros por la herramienta RTG-Command

$java -jar RTG.jar vcfmerge varscan_snp_filter.vcf.gz varscan_indel_filter.vcf.gz -o merge.vcf

Y lo mismo para el SNVever.

$java -jar RTG.jar vcffilter -i SNVever.vcf -o SNVever_filter.vcf.gz -q 20 -d 8
$java -jar RTG.jar vcffilter -i SNVever.vcf.indel.filter.vcf -o SNVever.vcf.indel.filter.vcf.gz -q 20 -d 8

Se realizó la fusión de ambos ficheros

$java -jar RTG.jar vcfmerge SNVever_filter.vcf.gz SNVever.vcf.indel.filter.vcf.gz -o SNVever_merge.vcf.gz

6-Obtención de resultados estadísticos

Tras el uso de los distintos detectores se procedió a obtener el reporte estadístico de cada una de ellos a través de la herramienta RTG Command vcfstats.

Para ello se aplicó el siguiente comando para cada uno de los VCF generados por cada una de los detectores una vez realizado su filtrado , como se aprecia en la imagen 7.

$java -jar RTG.jar vcfstats detector.vcf

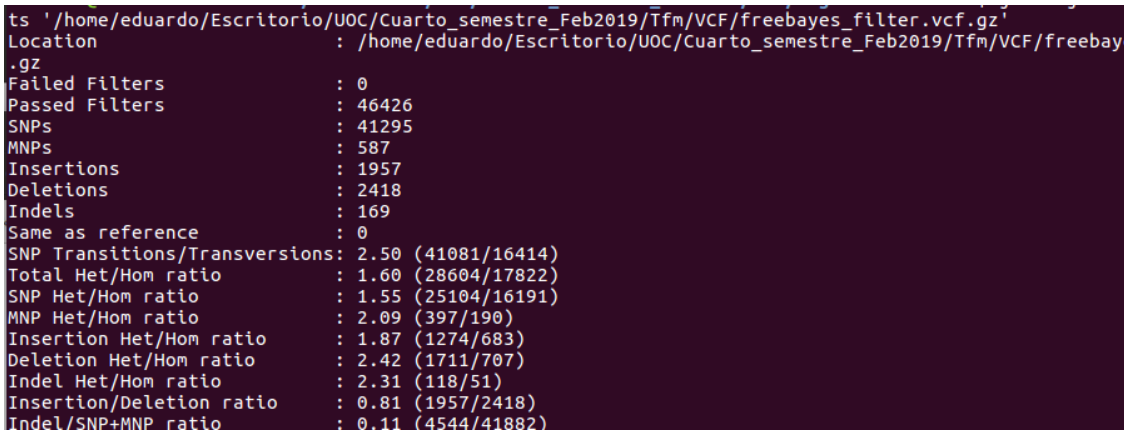


Imagen 7- Resultados estadísticos del VFC obtenido a partir del detector FreeBayes

Finalmente tras la ejecución del comando en cada uno de los detectores se obtuvieron los siguientes resultados (Tabla 1)

	FreeBayes	**Gatk-HC**	**SNVer**	**Samtools**	**VarScan**
N.º SNPs	41295	49609	38676	53705	33718
N.º Insercciones	1957	3074	1608	2336	1445
N.º Delecciones	2418	3954	1867	2550	1593
Ratio Het/Hom	1,60	1,64	1,41	1,23	1,23
Ratio SNP Het/Hom	1,55	1,56	1,21	1,25	1,19
Ratio Insercciones Het/Hom	1,87	1,88	Todos Het	0,77	1,52
Ratio Delecciones Het/Hom	2,42	2,98	Todos Het	1,25	1,94
Ratio Inserción/Delección	0,81	0,78	0,86	0,92	0,91

Tabla 1- Resultados estadísticos de cada uno de los detectores empleados para el análisis de variantes.

El detector que obtuvo un mayor número de SNP fue el Samtools con un total de 53705 frente a los 33718 de VarScan que fue el que menos obtuvo. Todos ellos superaron el umbral de los 25,000 SNP que ha visto que son normal para individuos de raza Europea. [https://www.biostars.org/p/187555/].

Respecto al número de insercciones y delecciones el detector que obtuvo mas variantes fue GATK-HC con un total de 7028 (3954 + 3074) doblando al que menor numero al detector con menor numero de variantes, VarScan 3038 (1445+1593).

Sobre los valores de ratio sobre Inserción/Delección, unos datos vemos que casi todos quedaron cercanos al valor 1, que es el ratio aceptable [https://gatkforums.broadinstitute.org/gatk/discussion/6308/evaluating-the-quality-of-a-variant-callset].

El ratio de variantes Heterocigotas frente a variantes Homocigotas se quedo con valores que oscilaron entre 1,64 y 1,23.

Finalmente observamos que una extraña anomalía en el detector SNVer donde únicamente se obtuvieron variantes de INDEL Heterocigotas.

7-Comparación frente al genotipo estándar.

Para la comparación con el genotipo de referencia usar el VCF de referencia y la opción de RTG command **vcfeval** que nos permite evaluar las called variant con nuestro baseline variant set (genotipo de referencia). Este genotipo de referencia fue descargado del siguiente ftp,
ftp://ftp-trace.ncbi.nlm.nih.gov/giab/ftp/data/NA12878/analysis/RTG_small_variants_011132014/

Primero es necesario realizar el formateo de los datos del genoma de referencia a un formato legible para la herramienta RTG.

$java -jar RTG.jar format -o genome_reference 'Homo_sapiens.GRCh37.dna.primary_assembly.fa.gz'

A continuación se realiza la comparación del VCF obtenido frente al VCF del genotipo de referencia, usando la opción –bed-regions para comparar unicamente los exones (ejemplo con detector Gatk-HC).

$java -jar RTG.jar vcfeval --baseline=singleton-illumina-wgs.vcf.gz --bed-regions=highconf_nextera_intersection_test.bed -c gatk.vcf_filter.vcf.gz -t genome_reference -o eval

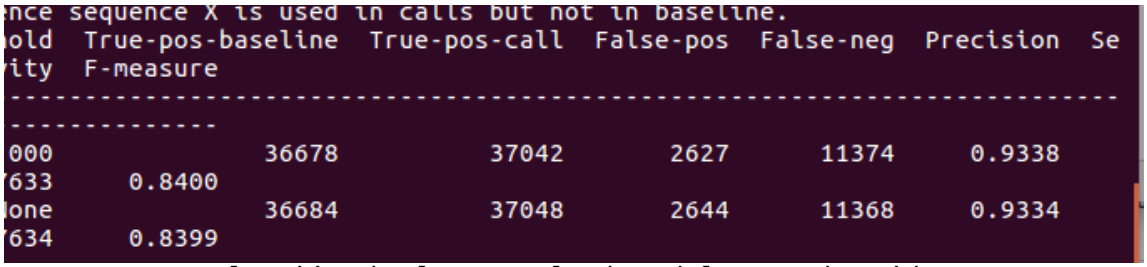


Imagen 8. Evaluación de los resultados del VCF obtenido por GATK-HC frente al genotipo de referencia.

Se repitió cada evaluación con los 5 detectores propuestos obteniendo los siguientes resultados (Tabla 2). En ello se exponen los conceptos de:

- Sensibilidad=TP/TP+FN
- Precisión=TP/(TP+FP)
- F-measure=2*Precisión*Sensibilidad/(Precisión+Sensibilidad)

Donde TP son verdaderos positivos, FN son Falsos negativos y FP son falsos positivos. El valor F-measure mide la precisión del detector.

BAM Q30

	TP-Baseline	TP-Call	Falsos Positivos	Falsos negativos	Precisión	Sensibilidad	F-measure
Free-Bayes	33257	33117	1621	14795	0.9533	0.7699	0.8520
Samtools	37168	37459	2044	10884	0.9483	0.7735	0.8520
GATK-HC	36678	37042	2627	11374	0.9338	0.7633	0.8400
VarScan	26943	27137	1391	21109	0.9512	0.5607	0.7055
SnvEver	29710	29910	2245	18342	0.9302	0.6183	0.7428

Tabla 2- Resultados de Sensibilidad y Especificad de los 5 detectores usados para la detección de Variantes obtenidos a partir del fichero BAM filtrado por mínimo de calidad de base de 30.

A la vista de los resultados estadísticos obtenidos se decidió que una de las causas que pudiera estar influyendo en algunos valores bajos de precisión o sensibilidad fuera que se fue demasiado severo a la hora establecer el primer filtro de calidad en el tratamiento del BAM crudo, que estableció en un valor inicial de 30. Se decide pues bajarlo hasta 10 y reproducir todos los pasos expuestos hasta llegar a la obtención de cada VCF. (Tabla 2).

BAM Q10

	TP-Baseline	TP-Call	Falsos Positivos	Falsos negativos	Precisión	Sensibilidad	F-measure
Free-Bayes	33282	33142	1640	14770	0.9528	0.6926	0.8022
Samtools	37174	37465	2062	10878	0.9478	0.7736	0.8519
GATK-HC	33751	34077	2353	14301	0.9354	0.7024	0.8023

VarScan	26983	27178	1421	21069	0.9503	0.5615	0.7059
SnvEver	29738	29939	2258	18314	0.9299	0.6189	0.7431

Tabla 3- Resultados de Sensibilidad y Especificad de los 5 detectores usados para la detección de Variantes obtenidos a partir del fichero BAM filtrado por mínimo de calidad de base de 10.

A la vista de los resultados obtenidos podemos observar que hemos obtenido buenos valores de especificidad, obteniendo un rango de valores de 0.93-0.95. Sin embargo los valores de sensibilidad han sido mas bajos de lo normal situandonos en una horquilla de sde 0.56 a 0.77. Esto implicaría que tenemos un mayor número de falsos negativos de lo normal, es decir, no se está detectando variantes que en principio presenta el genotipo estándar. Desde el punto de vista clínico sería importante reducir el número de falsos negativos y consecuentemente incrementar la sensibilidad ya que no detectar variantes es mas peligroso que detectar variantes que no esten presentes.

Para ello se ha establecido generar un VCF global que contenga todas las variantes de los detectores y que con ello se pueda incrementar el número de variantes que realmente están presentes en el genotipo estándar.

8-Generación del VCF global a partir de los VCF de los 5 detectores.

Se intento generar un VCF global con la herramienta RTG con la opción vcfmerge, pero se produjeron errores con la cabecera de los VCF, que se pudo corregir utilizando la opción -F que permite fusionar los archivos a pesar de que las cabeceras no coincidan.

El problema vino cuando se intento evaluar frente al genotipo de referencia, que mostraba que no se había suministrado un nombre de muestra.

Por recomendación del tutor se procedió a usar otro procedimiento para obtener los VCF fusionados. Para ello se descargo una versión más antigua del GATK tools, en concreto la 3,8. Se probo a usar el comando CombineVariants. Para ello había que pasarle por parámetros el genoma de referencia y cada uno de los VCF.

Finalmente se puedo realizar la ejecución del comando:

$java -jar GenomeAnalysisTK.jar -T CombineGVCFs -R Genome_bed.fa -V gatk.vcf_filter.vcf.gz -V samtools_filter.vcf.gz -o union_SNP_ALL.vcf

El problema fue que le llevo mas de 15 horas la ejecución del proceso y el output VCF generado no presentaba las variantes.

Se probo a utilizar el VCF fusionado obtenido a partir de la herramienta RTG command vcfmerge para usarlo para la selección de variantes a través de las Gatk tools SelectVariant pero produjo errores de cabecera malformada.

Se decidió intentar de nuevo crear el VCF fusionado a través de las Gatk tools corrigiendo las cabeceras de cada uno de los detectores de la siguiente manera.

$java -jar picard.jar FixVcfHeader I=detector.vcf O=fixed_detector.vcf

Posteriormente se realizo la unión de los 5 detectores a través de CombineVariants pero con la versión 3.8, ejecutando el siguiente comando:

$java -jar GenomeAnalysisTK.jar -T CombineVariants -R Homo_sapiens.GRCh37.dna.chromosome.1.fa --variant freebayes_filter_fixedHeader.vcf.gz --variant gatk.vcf_filter_fixedHeader.vcf.gz --variant mergesnever_fixedHeader.vcf.gz --variant samtools_filter_fixedHeader.vcf.gz --variant VarScan_merge_fixedHeader.vcf.gz -o Combine.vcf.gz

9-Evaluación del VCF global.

Analizando el fichero VCF generado se observó que en las variantes se notificada si había habido intersección, es decir, si la variante había sido detectado por los 5 detectores. Por lo tanto se procedió a realizar un filtro de este VCF donde hubiera intersección:

$grep 'set=Intersection' MergeCombineVariant.vcf | gawk '{ print $1,$2,$3,$4,$5,$6,$7,$8,$9,$10}' OFS='\t'>VCF_inter.vcf && cat CabeceraVCF VCF_inter.vcf> VCF_intersect.vcf

En este pipeline se realiza la selección del primer detector y el volcado a un fichero intermedio al que posteriormente se le añade la Cabecera del fichero VCF y finalmente se obtiene el fichero VCF con una muestra.

Posteriormente se realiza la compresión y el índice:

$java -jar RTG.jar bgzip VCF_intersect.vcf
$bcftools index -t VCF_intersect.vcf.gz

Finalmente se proceso a través de la herramienta RTG vcfeval y se obtuvo los siguiente resultados:

$java -jar RTG.jar vcfeval --baseline=singleton-illumina-wgs.vcf.gz --bed-regions=highconf_nextera_intersection_test.bed -c VCF_intersect.vcf.gz -t genome_reference -o eval

	TP-Baseline	TP-Call	Falsos Positivos	Falsos negativos	Precisión	Sensibilidad	F-measure
VCF_global	24648	24658	1384	23404	0.9469	0.5129	0.6654

Tabla 4- Resultados de Sensibilidad y Especificad de la unión de 5 detectores usados para la detección de Variantes obtenidos.

Como se puede apreciar los resultados no mejoraron los resultados obtenidos de manera individual, ya que aunque los resultados de precisión si que fueron cercanos a los mejores detectores los de sensibilidad bajaron considerablemente al verse incrementado el numero de falsos positivos.

A raíz de los resultados obtenidos se procedió a otra estrategia a la hora de obtener un VCF global.

En este caso se decidió establecer un criterio por el cual se seleccionarían los 3 detectores que por SNP hubieran tenido mejor resultado y 2 detectores que por INDEL a su vez hubieran tenido mejores resultados. Para ellos se realizaron varios que se exponen a continuación.

1-Selección de SNP e INDELs del genotipo estandar.

$java -jar GenomeAnalysisTK.jar -R Homo_sapiens.GRCh37.dna.chromosome.1.fa -T SelectVariants --variant singleton-illumina-wgs.vcf.gz --selectTypeToInclude INDEL -o singleton-illumina-wgs_INDEL.vcf.gz

$java -jar GenomeAnalysisTK.jar -R Homo_sapiens.GRCh37.dna.chromosome.1.fa -T SelectVariants --variant singleton-illumina-wgs.vcf.gz --selectTypeToInclude SNP -o singleton-illumina-wgs_SNP.vcf.gz

2-Selección de SNP e INDELs de cada uno de los detectores:

Previamente a este paso se observo problemas con las cabeceras así que se extrajo las cabeceras iniciales antes del procesado con SelectVariants y a través del siguiente ejecución de la combinación de este pipeline y distintos comandos se pudo obtener el VCF de Indels y SNP que aceptara posteriormente la herramiente RTGcomand.

Obtención de VCF INDEL

$java -jar GenomeAnalysisTK.jar -R Homo_sapiens.GRCh37.dna.chromosome.1.fa -T SelectVariants --variant gatk.vcf_filter.vcf.gz --selectTypeToInclude INDEL -o FB_INDEL && grep -v '#' FB_INDEL>FB_INDEL_nc && cat Samtools_Header.vcf FB_INDEL_nc>Samtools_filter_INDEL.vcf

Compresión y creación del indice:

$java -jar RTG.jar bgzip Samtools_filter_INDEL.vcf
$bcftools index -t Samtools_filter_INDEL.vcf.gz

Obtención de VCF SNP

$java -jar GenomeAnalysisTK.jar -R Homo_sapiens.GRCh37.dna.chromosome.1.fa -T SelectVariants --variant gatk.vcf_filter.vcf.gz --selectTypeToInclude INDEL -o FB_SNP && grep -v '#' FB_SNP>FB_SNP_nc && cat Samtools_Header.vcf FB_SNP_nc>Samtools_filter_SNP.vcf

Compresión y creación del índice:

$java -jar RTG.jar bgzip Samtools_filter_SNP.vcf
$bcftools index -t Samtools_filter_SNPvcf.gz

Se repitió estos pasos para los 5 detectores y finalmente se evaluaron de manera individual por la herramienta RTGcommand.

A continuación se muestran los resultados obtenidos.

Evaluación de INDELS

	TP-Baseline	TP-Call	Falsos Positivos	Falsos negativos	Precisión	Sensibilidad	F-measure
Free-Bayes	2201	2204	757	2508	0.7443	0.4674	0.5742
Samtools	2286	2286	549	2423	0.8063	0.4855	0.6060
GATK-HC	2771	2796	1233	1938	0.6940	0.5884	0.6369
VarScan	1539	1542	464	3170	0.7687	0.3268	0.4586
Snver	791	793	1377	3918	0.3654	0.1680	0.2302*

Tabla 5- Resultados de Sensibilidad y Especificad de los 5 detectores usados para la detección de INDELs en comparación con el el genotipo estándar.

* Los valores de SNVer fueron realmente bajos, esto puede tener una explicación plausible y es que por alguna razón este detector unicamente genero variantes heterocigotas en caso de los INDELs.

Evaluación de SNPs

	TP-Baseline	TP-Call	Falsos Positivos	Falsos negativos	Precisión	Sensibilidad	F-measure
Free-Bayes	30247	30247	859	12746	0.9724	0.7035	0.8164
Samtools	34596	34596	2072	8397	0.9435	0.8047	0.8686
GATK-HC	33579	33579	2065	9414	0.9421	0.7810	0.8540
VarScan	25215	25215	1307	17778	0.9507	0.5865	0.7255

Snver	28723	28723	1262	14270	0.9579	0.6681	0.7872

Tabla 6- Resultados de Sensibilidad y Especificad de los 5 detectores usados para la detección de SNPs en comparación con el el genotipo estándar.

Después de los datos observado seleccionamos los 3 primeros detectores para hacer la unión.

$java -jar GenomeAnalysisTK.jar -T CombineVariants -R Homo_sapiens.GRCh37.dna.chromosome.1.fa --variant freebayes_filter_fixedHeader.vcf.gz --variant gatk.vcf_filter_fixedHeader.vcf.gz --variant samtools_filter_fixedHeader.vcf.gz -o Combine.vcf.gz

$grep 'set=Intersection' MergeCombineVariant.vcf | gawk '{ print $1,$2,$3,$4,$5,$6,$7,$8,$9,$10}' OFS='\t'>VCF_inter.vcf && cat CabeceraVCF VCF_inter.vcf> VCF_intersect.vcf
$java -jar RTG.jar bgzip VCF_intersect.vcf
$bcftools index -t VCF_intersect.vcf.gz

Obtención de los resultados tras la selección de los 3 mejores detectores (FreeBayes, Samtools y GATK) a través través de la herramienta RTG vcfeval.

$java -jar RTG.jar vcfeval --baseline=singleton-illumina-wgs.vcf.gz --bed-regions=highconf_nextera_intersection_test.bed -c VCF_intersect.vcf.gz -t genome_reference -o eval

	TP-Baseline	TP-Call	Falsos Positivos	Falsos negativos	Precisión	Sensibilidad	F-measure
VCF de los 3 mejores detectores	31853	31884	907	16199	0.9723	0.6629	0.7883

Tabla 7- Resultados de Sensibilidad y Especificad de la unión de los 3 detectores (Freebayes, Samtools y GATK_HC) usados para la detección de variantes en comparación con el el genotipo estándar.

Se aprecia como se mejoraron tanto los resultados de sensibilidad como de especificad frente a la combinación de los 5 detectores de forma global. Sin embargo no ha habido una mejora en Sensibilidad frente al mejor detector que fue el de Samtools con un valor de 0.7735 .

Finalmente se realizo una nueva consulta, haciendo que se seleccionaran unicamente variantes que hubieran sido detectadas en 2 o mas detectores, es decir, las variantes que solo fueran detectadas por un detector se despreciarían.

Para ello se ejecuto los siguiente comandos.

```
$grep -v 'set=variant\t' Combine3.vcf |grep -v 'set=variant2'| grep -v 'set=variant3' | grep -v '#' |gawk '{ print $1,$2,$3,$4,$5,$6,$7,$8,$9,$10}' OFS='\t'>VCF_inter2.vcf && cat CabeceraVCF VCF_inter2.vcf> VCF_intersect2.vcf
$java -jar RTG.jar bgzip VCF_intersect2.vcf
$bcftools index -t VCF_intersect2.vcf.gz
```

Y se obtuvieron los siguientes resultados.

	TP-Baseline	TP-Call	Falsos Positivos	Falsos negativos	Precisión	Sensibilidad	F-measure
VCF de los 3 mejores detectores	36652	36987	2799	11400	0.9296	0.7628	0.8380

Tabla 8- Resultados de Sensibilidad y Especificad de la unión de los 3 detectores (Freebayes, Samtools y GATK_HC) escogiendo variantes que solo hubieran sido detectadas por mas de un detector.

Vemos que mejoraron los resultados de sensibilidad pero disminuyeron los resultados de especificidad.

A continuación se muestra una tabla resumen con la clasificación de los detectores en función de la evaluación global de SNP e INDEL.

Clasificación Detector	**Sensibilidad**	**Especificidad**
1	Samtools (0.7735)	VCF Global 3 mejores detectores (0.9723)
2	Freebayes (0.7699)	FreeBayes (0.9533)
3	VCF 3 mejores detectores, intersección en 2. (0.7628)	VarScan (0.9512)
4	GATK-HC (0.7633)	Samtools (0.9483)
5	VCF Global 3 mejores dectores (0.6629)	GATK-HC (0.9338)
6	SnVer (0.6183)	SNVer (0.9302)
7	VarScan (0.5607)	VCF 3 mejores detectores, intersección en 2 (0.9296)

A la vista de los resultados se entiende que las mejores herramientas para la detección de variantes sin hacer la fusión de los distintos detectores fueron Samtools, Freebayes y GATK-HC con valores de sensibilidad siempre superiores a 0,7. Mientras que se colaron

Los mejores detectores de los propuestos a la hora de analizar datos de Exoma serían los de FreeBayes, GATK-HC y Samtools/Bcftools. Estos detectores obtuvieron mejores resultados tanto para la evaluación de INDELs como la de SNPs.

Por otro lado se ha visto que uno de los objetivos planteados que era la unión de todos los detectores para la obtención de un VCF global, no mejoro los resultados de los detectores por separados al producir un incremento en el número de falsos negativos.

Tras estos resultados se optó por hacer una mejor selección de los detectores en función del tipo de variante SNP o INDELs y nos volvimos a quedar con los tres detectores arriba mencionados. Se generó un VCF que contuviera la unión de los tres detectores y en este caso si que vimos una mejora en el rendimiento en cuanto a Especificidad (0.9723 vs 0.9533) però hubo también una reducción en Sensibilidad (0.7735 vs 0.6629) debida nuevamente al incremento en el número de falsos negativos. Finalmente se hizo una última prueba para ver si la unión de los 3 mejores detectores donde hubiera intersección al menos de dos detectores mejoraría los resultados y se pudo apreciar que se incrementaron los resultados de sensibilidad (0.7628 vs 0.6629) aunque se redujeron los de especificidad (0.9296 vs 0.9723).

3. Conclusiones

La primera conclusión tras el trabajo realizado es que en función de los tratamientos que se realicen durante cada una de las etapas de procesamiento de FASTQ, BAM o VCF van a alterar los resultados, finales incrementando o dismuyendo el número de variantes en nuestros fichero finales.

En caso de buscar una mayor Sensibilidad en la análisis de variantes SNP e INDELs, el detector que ofrece mejores resultados es Samtools, seguido de Freebayes. Aunque también se encontró una alta sensibilidad en aquellas variantes comúnes en, por lo menos, dos dectectores al realizar la fusión de los ficheros VCF procedentes de los tres mejores detectores (Samtools, Freebayes y GATK-HC).

Respecto a la Especifidad, los mejores valores se obtienen tras realizar la unión de los tres mejores detectores (Samtools, Freebayes y GATK-HC). Por tanto, cuando se requieran encontrar variantes de un modo más específico, se recomienda realizar esta combinación para mejores los resultados de Especificidad.

En general, se han podido conseguir todos los objetivos fijados inicialmente en este trabajo final de máster. Sin embargo, para llevarlos a cabo se han tenido que cambiar ciertos planteamientos iniciales, como por ejemplo, la sustitución del 5º detector de variantes a SNVer incialmente fijada para el Deepvariant.

El principal escollo a la hora de la consecución de este TFM ha sido que al utilizar múltiples herramientas tanto para el procesamiento de los distintos archivos como para la detección de variantes que generaban incompatibilidades que rompían las pipelines asignadas inicialmente.

Finalmente exponer que este trabajo está enfocado para ser un punto de partida dentro de un análisis bioinformático de muestras FastQ procedentes de secuenciación de Exoma a través de la tecnología de secuenciación de Illumina. Sin embargo para llegar a conclusiones más definitorias habría que probarlo sobre un mayor número de muestras que ya hubieran sido validadas por otros centros y ver si las variantes detectadas funcionan con valores de sensibilidad y especificidad semejantes a los observados frente a la muestra control NA12878.

4. Glosario

SNP- Single Nucleotide Polymorphism. Polimorfismo de una sola base.
INDEL-Insercción o Delección.
SAM-Sequence Alignment Map. Fichero que contiene las secuencias alineadas.
BAM-Versión Binaria de SAM.
VCF-Variant Calling Format. Fichero que contiene los datos de variantes.
FASTQ-Formato de fichero que contiene los datos de la secuenciación de una muestra.
NGS-Next Generation Sequencing. Técnica de secuenciación massiva.
Secuenciación masiva- Tecnología que permite secuenciar el genoma con mayor cantidad de información.
Detector-Software o Herramienta bioinformática que procesa fichero de alineamientos para obtener ficheros VCF.
Exoma- Parte del genoma que es codificante de proteína
Variantes-En una misma región del genoma puede haber distintas copias
Zonas de captura- Regiones dentro del genoma que son seleccionadas para analizarse, ejemplo un exon dentro de un gen.
Pipeline- Consiste en ir transformando un flujo de datos en un proceso comprendido por varias fases secuenciales, siendo la entrada de cada una la salida de la anterior.
TFM-Trabajo Fin de Master.

5. Bibliografía

1- Hwang S, Kim E, Lee I, Marcotte EM. Systematic comparison of variant calling pipelines using gold standard personal exome variants. Sci Rep. 2015 Dec 7;5:17875. doi: 10.1038/srep17875.

2- Pabinger S, Dander A, Fischer M, Snajder R, Sperk M, Efremova M, Krabichler B, Speicher MR, Zschocke J, Trajanoski Z. A survey of tools for variant analysis of next-generation genome sequencing data. Brief Bioinform. 2014 Mar;15(2):256-78. doi: 10.1093/bib/bbs086. Epub 2013 Jan 21.

3-Farkas C, Fuentes-Villalobos F, Rebolledo-Jaramillo B, Benavides F, Castro AF, Pincheira R.Streamlined computational pipeline for genetic background characterization of genetically engineered mice based on next generation sequencing data. BMC Genomics. 2019 Feb 12;20(1):131. doi: 10.1186/s12864-019-5504-9.

4- Mu W, Lu HM, Chen J, Li S, Elliott AM. Sanger Confirmation Is Required to Achieve Optimal Sensitivity and Specificity in Next-Generation Sequencing Panel Testing. J Mol Diagn. 2016 Nov;18(6):923-932. doi: 10.1016/j.jmoldx.2016.07.006. Epub 2016 Oct 6.

5- Li H, Durbin R. Fast and accurate short read alignment with Burrows-Wheeler transform. Bioinformatics. 2009 Jul 15;25(14):1754-60. doi: 10.1093/bioinformatics/btp324. Epub 2009 May 18.

6- Jonathan Pevsner. Bioinformatics and Functional Genomics.

7-Koboldt DC, Larson DE, Wilson RK.Using VarScan 2 for Germline Variant Calling and Somatic Mutation Detection. Curr Protoc Bioinformatics. 2013 Dec;44:15.4.1-17. doi: 10.1002/0471250953.bi1504s44.

8-MARTIN, Marcel. Cutadapt removes adapter sequences from high-throughput sequencing reads. **EMBnet.journal**, [S.l.], v. 17, n. 1, p. pp. 10-12, may 2011. ISSN 2226-6089.. doi:https://doi.org/10.14806/ej.17.1.200.

9-James T. Robinson, Helga Thorvaldsdóttir, Wendy Winckler, Mitchell Guttman, Eric S. Lander, Gad Getz, Jill P. Mesirov. **Integrative Genomics Viewer**. Nature Biotechnology 29, 24–26 (2011).

10-Helga Thorvaldsdóttir, James T. Robinson, Jill P. Mesirov**. Integrative Genomics Viewer (IGV): high-performance genomics data visualization and exploration**. Briefings in Bioinformatics 14, 178-192 (2013).

11-Li H.*, Handsaker B.*, Wysoker A., Fennell T., Ruan J., Homer N., Marth G., Abecasis G., Durbin R. and 1000 Genome Project Data Processing Subgroup (2009) The Sequence alignment/map (SAM) format and SAMtools. Bioinformatics, 25, 2078-9. [PMID: 19505943]

12-* Li H. A statistical framework for SNP calling, mutation discovery, association mapping and population genetical parameter estimation from sequencing data. Bioinformatics. 2011 Nov 1;27(21):2987-93. Epub 2011 Sep 8. [PMID: 21903627]

13-Andrews S. (2010).FastQC: a quality control tool for high throughput sequence data. Available online at: http://www.bioinformatics.babraham.ac.uk/projects/fastqc

14-Aaron R. Quinlan Ira M. Hall. BEDTools: a flexible suite of utilities for comparing genomic features .Bioinformatics, Volume 26, Issue 6, 15 March 2010, Pages 841–842, https://doi.org/10.1093/bioinformatics/btq033

15-https://gatkforums.broadinstitute.org/gatk/discussion/12612/running-haplotypecaller-in-gatk. 30/04/19.

16-Pircard Tools. http://broadinstitute.github.io/picard/ 05/05/19.

17-Wei Z, Wang W, Hu P, Lyon GJ and Hakonarson H. SNVer: a statistical tool for variant calling in analysis of pooled or individual next-generation sequencing data, Nucleic Acids Research 2011, doi: 10.1093/nar/gkr599. [PMID: 21813454]

18-Garrison E, Marth G. Haplotype-based variant detection from short-read sequencing. *arXiv preprint arXiv:1207.3907 [q-bio.GN]* 2012

19**-** Koboldt, D., Zhang, Q., Larson, D., Shen, D., McLellan, M., Lin, L., Miller, C., Mardis, E., Ding, L., & Wilson, R. (2012). VarScan 2: Somatic mutation and copy number alteration discovery in cancer by exome sequencing *Genome Research* DOI: 10.1101/gr.129684.111

20-The Genome Analysis Toolkit: a MapReduce framework for analyzing next-generation DNA sequencing data McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, DePristo MA, 2010 *GENOME RESEARCH 20:1297-303*