Universitat Oberta de Catalunya

uoc.edu

Impacto en la clínica del uso de diferentes genomas de referencia en la llamada de variantes

Pedro Salvador Escribano

Máster universitario en Bioinformática y bioestadística UOC-UB

Área de Genómica comparativa y clínica

Consultor: José Luis Villanueva Cañas

Profesores responsables de la asignatura: Carles Ventura Royo, Marc Maceira Duch.

Junio 2020

FICHA DEL TRABAJO FINAL

Título del trabajo:	*Impacto en la clínica del uso de diferentes genomas de referencia en la llamada de variantes*
Nombre del autor:	*Pedro Salvador Escribano*
Nombre del consultor/a:	José Luis Villanueva Cañas
Nombre del PRA:	Carles Ventura Royo, Marc Maceira Duch
Fecha de entrega (mm/aaaa):	06/2020
Titulación:	*Máster universitario en Bioinformática y bioestadística UOC-UB*
Área del Trabajo Final:	*Genómica comparativa y clínica*
Idioma del trabajo:	*Castellano*
Palabras clave	*llamada de variantes, genomas de referencia humanos, next generation sequencing*

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

El abaratamiento de los costes de secuenciación por NGS, unido a la rapidez con la que puede llevarse a cabo este tipo de técnica, ha permitido su utilización como técnica diagnóstica. Durante la llamada de variantes, se buscan diferencias entre la secuencia genética del individuo analizado y un genoma de referencia. La elección de éste genoma de referencia puede condicionar los resultados obtenidos. Dado que se han descrito numerosos SNVs que se consideran factores de riesgo para diversas patologías, la precisión en su detección cobra una importancia crucial. Este trabajo analiza la relevancia clínica de las posibles diferencias en la identificación de variantes causada por la elección de uno u otro genoma de referencia. Para ello, se llevó a cabo la llamada de variantes en muestras procedentes de secuenciación de exomas por NGS, utilizando los genomas de referencia hg19 y hg38, tras lo que se cuantificó las diferencias obtenidas en cada uno de los análisis y se clasificó aquellas variantes cuya detección difería con el uso de los diferentes genoma de referencia atendiendo a diferentes criterios, tras lo que se concluyó que es recomendable el uso de hg38 como genoma de referencia durante la llamada de variantes con fines clínicos.

Abstract (in English, 250 words or less):

The lowering of the sequencing costs by NGS, together with the speed with which this type of technique can be carried out, has allowed its use as a diagnostic technique. During the variant calling, differences are sought between the genetic sequence of the analyzed individual and a reference genome. The choice of this reference genome can condition the results obtained. Since numerous SNVs have been described that are considered risk factors for various pathologies, the accuracy of their detection is of crucial importance. This work analyzes the clinical relevance of the possible differences in the identification of variants caused by the choice of one or another reference genome. For this, the variant calling was carried out in samples from NGS exome sequencing, using the reference genomes hg19 and hg38, after which the differences obtained in each of the analysis were quantified and variants whose detection differed with the use of the different reference genomes were classified according to different criteria, after which, it was concluded that the use of hg38 as a reference genome is recommended during the variant calling for clinical purposes.

Índice

Índice de tablas

Índice de figuras

1. Introducción

1.1. Proyecto Genoma Humano y genomas de referencia

Con el primer borrador de la secuencia del genoma humano, publicado en 2001, se reportó que las diferencias genéticas entre dos individuos tomados al azar son de apenas el 0.1% del genoma[1,2]. Este proyecto, que tuvo un coste económico de más de 3000 millones de dólares, tuvo su origen en 1990 y no fue finalizado hasta el año 2003.

Uno de los principales objetivos de este proyecto fue la obtención de una secuencia consenso de referencia del genoma humano, pues la obtención de esta secuencia permitiría su comparación con posteriores secuencias de ADN obtenidas en diferentes estudios genéticos, pudiendo utilizarse para diversas metas como el ensamblado de genomas individuales, el estudio de la variabilidad genética natural o la identificación de las variantes genéticas causantes de enfermedades. Hoy en día, estas secuencias consenso, conocidas como genomas de referencia, son herramientas esenciales tanto en la investigación como en aplicaciones clínicas. Dada la existencia de variabilidad genética en la población, estos genomas de referencia no corresponden al ensamblado de la secuencia genética de un único individuo, sino que son secuencias consenso que derivan del estudio del ADN de diferentes voluntarios. Según la información proporcionada por el propio genome reference consortium, entorno al 70% de la secuencia de la última versión del genoma disponible proviene de un único individuo, otro 23% está compuesto por los genomas de 10 individuos y el 7% restante ha sido obtenido a partir de los genomas de más de 50 individuos[3].

Debido a su naturaleza, los genomas de referencia tienen una serie de limitaciones técnicas. Como se dijo previamente, estos genomas se obtienen por un consenso entre los genomas de varios individuos, por lo que en ellos no está reflejada la variabilidad total que existe en la especie humana. Estos genomas de referencia tendrán por tanto una menor utilidad cuando se estén utilizando para comparar el ADN de individuos que no compartan acervo genético con los utilizados para la elaboración del genoma de referencia utilizado, no pudiendo diferenciar en estos casos si las diferencias encontradas se deben a variaciones normales en la población de pertenencia del individuo analizado o son indicativas de variantes patológicas.

1.2. Secuenciación de próxima generación

El desarrollo de nuevas técnicas de secuenciación ha permitido reducir enormemente tanto el coste como el tiempo necesario para la secuenciación de un genoma completo. La secuenciación

de próxima generación ("next generation sequencing" (NGS)) o secuenciación masiva paralela permite la secuenciación genética de individuos con resolución de un solo nucleótido. Así mismo, esta técnica permite la detección de todos los tipos de variación genómica de manera simultánea, incluyendo mutaciones puntuales o variantes de nucleótido único, pequeñas inserciones y deleciones y variantes estructurales como grandes deleciones, duplicaciones, inversiones y translocaciones. La rapidez con la que puede obtenerse esta información a través de esta técnica, así como la reducción de los costes de la misma en los últimos años, han hecho posible su incorporación a las aplicaciones clínicas.

Aunque existen diferentes tecnologías de secuenciación por NGS actualmente y estas difieren en varios aspectos, todas ellas siguen un esquema de trabajo similar. En primer lugar, el ADN que se desea secuenciar es fragmentado, tras lo que se incorporan unas secuencias adaptadoras en los extremos mediante ligamiento, formando lo que se conoce como librería de ADN. En una siguiente etapa, se produce la amplificación clonal de los fragmentos de la librería. La secuenciación es entonces llevada a cabo sobre estos fragmentos amplificados. Gracias a la amplificación clonal, se consigue que una misma base sea secuenciada en varias ocasiones, aumentando así el número de lecturas sobre la misma, lo que se conoce como profundidad de cobertura.

1.3. Uso de NGS en la búsqueda de variantes

A partir de datos obtenidos de NGS, puede realizarse la búsqueda de variantes genéticas que pueden estar implicadas en diferentes patologías o estar relacionadas con la respuesta diferencial a fármacos de los individuos[4-6]. Este proceso se conoce como llamada de variantes[7,8].

La detección de variantes a partir de datos procedentes de NGS se lleva a cabo mediante la comparación de secuencia de ADN del individuo con un ADN de referencia. Se realiza un proceso mediante el que las lecturas cortas procedentes de la plataforma de secuenciación se alinean contra la cadena larga del genoma de referencia[9]. Dada la naturaleza de este proceso, los resultados dependen de la calidad de este alineamiento, por lo que defectos en la alineación pueden desencadenar la aparición de falsos positivos mientras que la falta de alineamiento de secuencias produciría falsos negativos. En la práctica, se suele utilizar una de las dos versiones más recientes del genoma humano de referencia: GRCh37/hg19 (Genome Research Consortium human build 37/human genome build 19) o GRCh38/hg38 (Genome Research Consortium human build 38/human genome build 38). Normalmente se recomienda el uso de la versión más reciente[10], pero la existencia previa de datos con las que hacer comparativas en versiones

anteriores hace que éstas resulten en ocasiones atractivas para los investigadores, a pesar de que se ha demostrado que los resultados obtenidos tienen una menor precisión[11].

Aunque la precisión con la que se detectan las variantes es muy alta, la elección del genoma de referencia puede condicionar los resultados obtenidos en el análisis, pudiendo haber discordancias dependiendo del escogido[12].

Hasta la fecha no se ha analizado el posible impacto en la clínica de las discordancias obtenidas durante la llamada de variantes, debidas a la elección de uno u otro genoma de referencia.

Variantes de nucleótido único

Los polimorfismos puntuales que afectan a una única base de la secuencia de ADN se clasifican en función de su frecuencia en la población, considerándose como variantes de nucleótido único ("sinle nucleotide variant" (SNV)) o SNP (*Single Nucleotide Polymorphism)* si su presencia es mayor al 1% o mutaciones puntuales si es esta es menor. Este tipo de variación representa el tipo mayoritario en el genoma humano, considerándose que aparecen como promedio cada 1300 bases a lo largo del mismo.

La mayoría de las mutaciones causantes de enfermedades son mutaciones exónicas no sinónimas en SNV. Estas mutaciones suelen afectar a la función de la proteína codificada, por lo que son las más estudiadas. Pese a ello, existe evidencia de que al menos el 80% del genoma humano tiene alguna funcionalidad, al haberse demostrado que tiene cierta capacidad de transcripción o al menos muestra interacción con proteínas, por lo que mutaciones en áreas no codificantes también pueden estar implicadas en procesos como la unión de ARN de interferencia, la alteración de los patrones de metilación del ADN, la alteración de la estructura local de ARNs no codificantes o la regulación génica a través de *enhancers* y promotores entre otros[13,14].

Los sistemas de detección de este tipo de variantes se basan en modelos bayesianos, con los que se calcula la probabilidad de presencia de cada nucleótido en cada posición según el número de lecturas independientes realizadas para el mismo. Para su buen funcionamiento, estos programas requieren de una profundidad de cobertura suficiente para poder realizar este tipo de cálculos estadísticos, así como de una baja tasa de error en la lectura. Aunque los errores en la secuenciación se producen generalmente al azar, las diferentes plataformas de secuenciación sesgan la aparición de este tipo de errores de diferente manera. Por ejemplo, la plataforma Illumina, induce la aparición de errores hacia el final de las lecturas. Un análisis de la calidad de las lecturas y el filtrado de aquellas que no alcanzan la calidad necesaria para proporcionar una

fiabilidad aceptable es, por tanto, un requisito previo indispensable para llevar a cabo la llamada de variantes con datos procedentes de NGS.

Se han descrito numerosos SNVs que se consideran factores de riesgo para diferentes patologías, por lo que la precisión en la detección de estos SNVs cobra especial relevancia.

Inserciones y deleciones

Las pequeñas inserciones y deleciones *(indels)* son más difícilmente identificables a partir de datos procedentes de NGS debido a la menor longitud de las lecturas que generan este tipo de técnicas. Un caso especial es la de variantes de ganancia o pérdida de una única base, cuya tasa de error en la detección es mayor que en el caso de los SNV, pues su alineamiento con los genomas de referencia suele producirse de peor manera, produciendo una elevada tasa de falsos positivos.

Variantes de número de copia e inversiones grandes

Otro tipo de variación se produce cuando una secuencia de determinada longitud se repite secuencialmente un número variable de veces entre los diferentes individuos. El alineamiento de lecturas que comprenden estas regiones del genoma es muy proclive a producirse de forma errónea, por lo que se dificulta su detección. Este tipo de variación, junto con las grandes inversiones comparten un requisito para su correcta detección mediante alineamiento con un genoma de referencia. Se requiere que las lecturas tengan una longitud mayor a la del área de la variación. Por ello, las técnicas utilizadas para su detección suelen ser diferentes, por lo que no serán discutidas en este trabajo.

Diagnóstico molecular mediante NGS

Aunque la tecnología NGS está principalmente orientada a la investigación, su uso en medicina está extendiéndose cada vez más, tanto en su aplicación diagnóstica, para la detección de variantes patogénicas en enfermedades raras o de variantes de riesgo en enfermedades hereditarias, como en medicina preventiva, utilizándose, por ejemplo, en estudios farmacogenéticos que pretenden comprender la respuesta diferencial a fármacos de los individuos.

Hay diferentes técnicas que pueden llevarse a cabo para realizar este tipo de estudios, por lo que para cada caso habrá que valorar la idoneidad de una sobre las otras, ya que cada una ofrece una serie de ventajas e inconvenientes sobre las demás[15].

Secuenciación dirigida

Esta estrategia se basa en la secuenciación de una proporción del genoma conocida y su alineamiento con un genoma de referencia. La secuenciación de los loci de interés puede facilitar la interpretación de los resultados obtenidos, teniendo la ventaja de minimizar la posibilidad de hallazgos no relacionados con la prueba inicial. Este tipo de estrategia es adecuada cuando la enfermedad de la que se sospecha puede estar causada por mutaciones en múltiples genes conocidos. Además, con esta estrategia se reduce considerablemente el coste de la secuenciación al ser necesario secuenciar únicamente una pequeña fracción del genoma. Sin embargo, hay una serie de inconvenientes, pues los paneles utilizados deben ser constantemente actualizados al ampliarse el conocimiento sobre nuevos genes que puedan estar relacionados con la enfermedad, además de no ser una técnica adecuada cuando se trata de enfermedades raras, pudiendo ocasionar retrasos en su diagnóstico[16].

Secuenciación del exoma

El exoma comprende la porción codificante del genoma humano. Aunque su extensión es mucho mayor que la de un panel de genes, sigue siendo una proporción relativamente pequeña del genoma (entorno al 1,2% del mismo[17]), por lo que su secuenciación siempre será más barata que la del genoma completo. Una de las ventajas de la utilización de la secuenciación del exoma como técnica diagnóstica es la estandarización, pues no requiere de su actualización basada en nuevos descubrimientos y puede ser utilizada para multitud de enfermedades. Esto es especialmente beneficioso en el caso de las enfermedades raras y de manifestaciones anómalas de diferentes enfermedades en los casos en los que el fenotipo clínico del paciente no se corresponde con el fenotipo estándar asociado a la enfermedad[18]. Además, se ha demostrado que la secuenciación del exoma cubre más del 98% de las mutaciones identificadas en paneles de secuenciación dirigida[19].

Secuenciación de genoma completo

La secuenciación del genoma completo tiene la evidente ventaja de ofrecer el acceso a zonas reguladoras del genoma no expresadas, cuya mutación puede alterar la expresión génica. A cambio, el coste de secuenciación asciende notablemente. Aunque este coste puede ser asumible en la actualidad, hay otros inconvenientes asociados a esta estrategia, pues la interpretación de los datos generados se dificulta enormemente a la hora de realizar un diagnóstico, siendo dificil en ocasiones distinguir las variantes patogénicas de las no patogénicas, por lo que, usualmente, suele ser conveniente centrarse en la interpretación de

mutaciones ya conocidas. Dado que la distribución de las mutaciones conocidas causantes de enfermedad, se da mayoritariamente en la porción codificante del genoma, sería por tanto más eficaz la utilización de técnicas dirigidas en un comienzo y solo utilizar las técnicas más complejas en caso de falta de diagnóstico[20,21].

2. Objetivos

- Llevar a cabo la llamada de variantes a partir de datos de NGS utilizando los genomas de referencia HG19 y HG38.
- Analizar las diferencias entre los análisis llevados a cabo con ambas versiones del genoma.
- Evaluar el potencial impacto en la clínica de las variantes que difieran en su detección con el uso de cada genoma de referencia.

3. Material y métodos

3.1. Flujo de trabajo

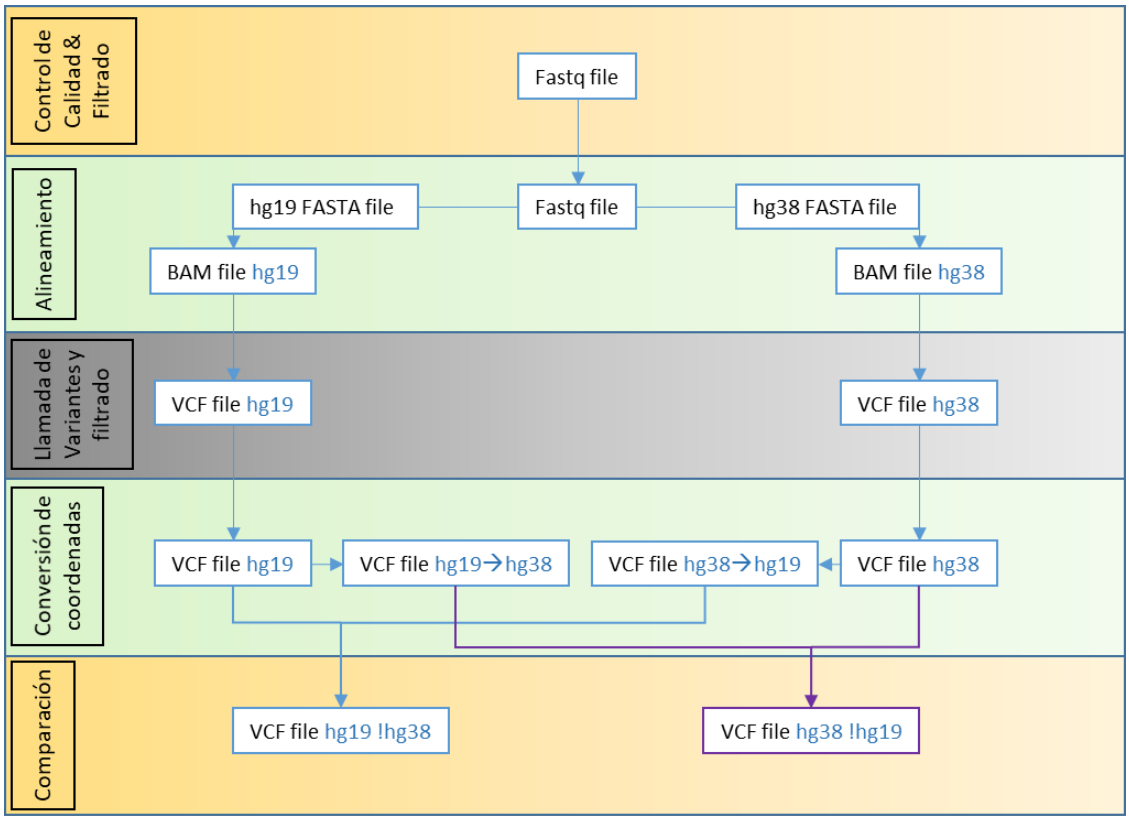


Figura 1 Flujo de trabajo realizado

3.2. Datos utilizados

Los datos utilizados en este trabajo han sido exomas secuenciados en formato FASTQ de pacientes con enfermedades metabólicas diagnosticadas. Se han obtenido los exomas de un total de cuatro pacientes. Esta información ha sido proporcionada por el Hospital Clínic de Barcelona. Para garantizar el anonimato de los pacientes, los datos han tienen una codificación numérica para cada uno de los pacientes, separando por tanto cualquier elemento que pudiera relacionar estos datos con los pacientes a partir de los cuales han sido obtenidos.

Nº Paciente (Código)	**Read 1**	**Read 2**
190830326_S6	190830326_S6_R1_001.fastq	190830326_S6_R2_001.fastq
190831050_S7	190831050_S7_R1_001.fastq	190831050_S7_R2_001.fastq
190837094_S7	190837094_S7_R1_001.fastq	190837094_S7_R2_001.fastq
202713578_S1	202713578_S1_R1_001.fastq	202713578_S1_R2_001.fastq

Tabla 1 Relación de muestras utilizadas en el estudio

3.3. Control de calidad.

Antes de iniciar un estudio bioinformático, debe llevarse a cabo un análisis previo de los datos de partida, con el fin de conocer si estos son susceptibles de ser utilizados o requieren de algún tipo de procesado o filtrado previo con el fin de hacerlos adecuados para los fines que se persiguen en el estudio. En datos procedentes de secuenciación masiva, es común llevar a cabo un análisis de la calidad de los mismos, con el fin de detectar posibles artefactos producidos fruto de la manipulación experimental que puedan interferir en posteriores etapas del análisis.

1.3.1. FastQC

El software elegido para llevar a cabo el control de calidad de los datos en formato FASTQ es FastQC[22]. Este programa ofrece una forma simple de hacer algunas comprobaciones de control de calidad en datos procedentes de secuenciación de alto rendimiento, proporcionando un conjunto modular de análisis que puede ser usado para dar una impresión rápida de si los datos tienen algún problema que debe tenerse en cuenta antes de realizar cualquier análisis posterior.

FastQC se puede ejecutar como una aplicación interactiva independiente para el análisis inmediato de pequeños números de archivos FASTQ, o en un modo no interactivo, más adecuado para su integración en el procesamiento sistemático de grandes cantidades de archivos.

Además de proporcionar un informe interactivo, FastQC también tiene la opción de crear una versión HTML de este informe para un registro más permanente. Este informe HTML también se puede generar directamente ejecutando FastQC en modo no interactivo. Junto al archivo HTML, FastQC proporciona un archivo ZIP que contiene los gráficos del informe como archivos separados, así como archivos de datos que permiten una evaluación más detallada y automatizada en caso de ser necesario.

El código para la ejecución del análisis de calidad con FastQC en línea de comandos es el siguiente:

```
fastqc <nombre_del_archivo>
```

Por tanto, para procesar todos los archivos FASTQ del directorio de trabajo se utilizaría el siguiente comando:

```
fastqc *.fastq
```

Como anexo adjunto se ofrece una explicación detallada de los módulos de análisis ofrecidos por FastQC.

1.3.2. MultiQC

Para una mejor comprensión de los resultados, los controles de calidad generados por FastQC son posteriormente procesados por MultiQC[23], un software que permite agregar los resultados de análisis bioinformáticos en muchas muestras en un solo informe.

MultiQC busca en un directorio dado los registros de análisis y compila un informe HTML. Es una herramienta de uso general, perfecta para resumir el resultado de numerosas herramientas bioinformáticas.

El código para la ejecución de MultiQC en línea de comandos es el siguiente:

```
multiqc .
```

Dependiendo del tipo de datos incluidos en los análisis, MultiQC ofrece un tipo de informe diferente, ofreciendo una visión simplificada de los análisis individuales producidos en cada archivo y ofreciendo una interfaz gráfica con todos los archivos incluidos en el análisis simultáneamente.

3.2. Filtrado de datos brutos

3.2.1. Trimmomatic

Usualmente, los datos procedentes de plataformas de secuenciación requieren un procesamiento previo antes de su análisis, con el fin de eliminar restos de adaptadores presentes en las secuencias y de eliminar lecturas con calidades inferiores a la deseada, que no permitirían realizar análisis con suficiente confianza. Trimmomatic[24] es una herramienta que permite realizar estas operaciones para datos FASTQ procedentes de plataformas Illumina. El procesado de los datos se produce según el orden de introducción de las diferentes opciones de

procesado en la línea de comandos. Las opciones disponibles para esta herramienta pueden ser consultadas en el anexo (“Trimmomatic”).

El comando básico de ejecución para secuenciaciones paired end es el siguiente:

```
java -jar <ruta/a/trimmomatic-0.39.jar> PE \
<input 1> <input 2> \
<paired output 1> <unpaired output 1> \
<paired output 2> <unpaired output 2> \
<step 1> <step 2> <step…>
```

Los parámetros fijados para el procesado de las muestras del presente estudio fueron los siguientes (Se muestra como ejemplo una de las muestras procesadas):

```
java -jar trimmomatic-0.39/trimmomatic-0.39.jar PE \
190830326_S6_R1_001.fastq 190830326_S6_R2_001.fastq \
190830326_S6_R1_001_paired.fastq 190830326_S6_R1_001_unpaired.fastq \
190830326_S6_R2_001_paired.fastq 190830326_S6_R2_001_unpaired.fastq \
ILLUMINACLIP:trimmomatic-0.39/adapters/adaptadores.fa:2:30:10:2:keepBothReads \
LEADING:3 TRAILING:3 SLIDINGWINDOW:4:20 MINLEN:100
```

Según este comando, el procesado se produjo de la siguiente forma:

Se procesaron las lecturas de la muestra 190830326_S6 en modo PE, obteniendo como output, cuatro archivos, dos para R1 y dos para R2., paired y unpaired. Los paired contienen las lecturas (R1 o R2) cuya pareja también pasa el filtro. Los unpaired contiene reads en que solo uno de los dos (R1 o R2) pasa el filtro.

ILLUMINACLIP:trimmomatic-0.39/adapters/adaptadores.fa:2:30:10:2:keepBothReads especifica el archivo fasta que contiene los adaptadores (adaptadores.fa), y las condiciones de aplicación del filtro (2:30:10:2:keepBothReads).

LEADING:3 indica eliminar las primeras bases de cada lectura con una calidad inferior a 3.

TRAILING:3 indica eliminar las últimas bases de cada lectura con una calidad inferior a 3.

SLIDINGWINDOW:4:20 corta lecturas cuando 4 bases seguidas bajan de calidad 20.

MINLEN:100 indica la eliminación de lecturas con una longitud menor que 100.

El archivo adaptadores.txt fue obtenido tras la concatenación de todos los archivos que especifican adaptadores Truseq para paired end utilizando el comando:

```
cat TruS*PE > adaptadores.txt
```

3.3. Genomas de referencia:

Tras el preprocesado de los archivos FASTQ procedentes de la plataforma de secuenciación como prerrequisito para su análisis, se procedió a la obtención de los genomas de referencia humanos GRCh37 (hg19) y CRCh38 (hg38). La descarga de las secuencias en formato fasta comprimido y su posterior descompresión se realizó de los servidores de la Universidad de California en Santa Cruz (UCSC) a través de los siguientes comandos:

Descarga:

```
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg19/bigZips/hg19.fa.gz
wget http://hgdownload.cse.ucsc.edu/goldenPath/hg38/bigZips/hg38.fa.gz
```

Descompresión:

```
gzip -d hg19.fa.gz
gzip -d hg38.fa.gz
```

3.4. Alineamiento:

3.4.1. Burrows-Wheeler Aligner (BWA)

El primer paso en el proceso de llamada de variantes es el alineamiento con los genomas de referencia. Mediante este proceso, se lleva a cabo un mapeo de las lecturas contenidas en los archivos FASTQ frente al genoma de referencia, de forma que estas puedan ser comparadas posteriormente con la secuencia consenso. Existen multitud de herramientas informáticas que pueden ser utilizadas para llevar a cabo este proceso, pero en este trabajo, se ha decidido utilizar la herramienta Burrows-Wheeler Aligner (BWA)[25], un paquete de programas utilizados para mapear secuencias de baja divergencia contra un genoma de referencia grande, como el genoma humano. Este programa se compone de tres algoritmos: BWA-backtrack, BWA-SW y BWA-MEM. De estos, el utilizado para los objetivos del presente estudio ha sido BWA-MEM (maximal exact matches), que es el recomendado para consultas de alta calidad, ya que es más

rápido y preciso y tiene un mejor rendimiento que BWA-backtrack para las lecturas de 70-100 pb de las plataformas Illumina.

Tras su descarga, el primer paso a realizar sobre los genomas de referencia es su indexado, de forma que las lecturas procedentes de la plataforma de secuenciación puedan ser correctamente alineadas contra estos genomas de referencia. El comando utilizado para la ejecución de este indexado es el siguiente:

```
bwa index [-p prefix] [-a algoType] <in.db.fasta>
```

donde:

bwa index invoca el algoritmo,

[-p prefix] indica el prefijo que tendrán los archivos producidos,

[-a algoType] indica el algoritmo utilizado para el alineamiento,

<in.db.fasta> indica el archivo fasta a ser indexado.

Los parámetros fijados para el procesado de los genomas de referencia del presente estudio fueron los siguientes:

```
bwa index -p hg19bwaidx -a bwtsw hg19.fa

bwa index -p hg38bwaidx -a bwtsw hg38.fa
```

Se utilizó el algoritmo bwtsw, que es el indicado para el indexado de genomas completos.

3.4.2. SAMtools

Tras este indexado, se puede llevar a cabo el alineamiento de los archivos FASTQ con BWA-MEM. Este algoritmo genera un archivo SAM (Sequence Alignment Map), que solo será utilizado como intermediario en el proceso de análisis y cuyo volumen es muy alto, por lo que para ahorrar tiempo y memoria de disco durante el procesado, puede enlazarse este objeto con el siguiente programa utilizado, SAMtools[26], que es el programa de elección para la visualización y manipulación de archivos SAM. Este programa ha sido utilizado para generar el equivalente binario de un archivo SAM, un archivo BAM (Binary Alignment Map), que almacena los mismos datos en una representación binaria comprimida. Las principales utilidades de SAMtools son las siguientes:

- view: extrae el total de los alineamientos o una porción de ellos.

- sort: ordena el alineamiento desde el extremo izquierdo, generando un nuevo archivo en formato BAM.
- index: genera un fichero con extensión '.bai' con un índice de los alineamientos ordenados.
- faidx: genera un fichero con extensión '.fai' con un índice de los archivos fasta que permite un acceso eficiente a regiones arbitrarias dentro de las secuencias de referencia.

Para el alineamiento se utilizó el siguiente comando:

```
bwa mem db.prefix reads.fq [mates.fq] | samtools sort -o <output>
```

donde:

bwa mem invoca el algoritmo,

db.prefix es el prefijo indicado mediante el indexado del genoma de referencia,

reads.fq [mates.fq] es la pareja de archivos FASTQ de cada paciente.

|se utiliza para indicar que la salida de este comando se utiliza como entrada para el siguiente,

samtools sort invoca el algoritmo,

-o <output> indica el nombre del archivo BAM generado.

Los parámetros fijados para el procesado de las muestras del presente estudio fueron los siguientes (Se muestra como ejemplo una de las muestras procesadas):

```
bwa mem -M hg19bwaidx 190830326_S6_R1_001_paired.fastq
190830326_S6_R2_001_paired.fastq | samtools sort -o
190830326_S6_paired_hg19_sorted.bam
```

3.5. Llamada de variantes:

Tras efectuar los pasos previos requeridos, puede llevarse a cabo la llamada de variantes, que consiste en la búsqueda de diferencias entre las lecturas efectuadas en las muestras utilizadas y la secuencia consenso del genoma de referencia.

Existen dos tipos de llamadas de variantes, la germinal y la somática. La diferencia entre ellas radica en la referencia utilizada, pues mientras que, en la llamada de variantes germinales, el objetivo es la búsqueda de diferencias en el genoma con respecto a un genoma de referencia, durante la llamada de variantes somáticas, el objetivo es encontrar mutaciones de novo producidas en un individuo, por ejemplo, durante procesos cancerígenos, por lo que se comparará tejido cancerígeno con tejido sano del mismo individuo. En el caso de este estudio, llevamos a cabo la llamada de variantes del tipo germinal.

Este proceso puede ser llevado a cabo con multitud de herramientas informáticas y para este trabajo, se han escogido dos de las más ampliamente extendidas y validadas por la comunidad científica, Strelka2[27] y Genome Analysis Toolkit (GATK)[28] y como herramientas para la preparación de los archivos necesarios para su ejecución, se han utilizado SAMtools[26] y Picard[29].

3.5.1. SAMtools

Generación de archivos FAI:

Como requisito para la ejecución de strelka2, se requiere que en la misma carpeta donde se encuentre el archivo FASTA correspondiente al genoma de referencia, se encuentre un archivo FAI. Este archivo puede ser generado mediante SAMtools[26] mediante el siguiente comando (Se muestra los comandos necesarios para la creación de los archivos FAI para hg19 y hg38):

```
samtools faidx hg19.fa

samtools faidx hg38.fa
```

Indexado de archivos BAM:

Un segundo requisito es que el archivo BAM a ser procesado sea previamente indexado. Esta indexación se puede realizar mediante SAMtools con el siguiente comando (Se muestra como ejemplo una de las muestras procesadas):

```
samtools index 190830326_S6_paired_hg19_sorted.bam
```

3.5.2. Strelka2

Strelka2[27] permite realizar la llamada de variantes de forma rápida y precisa. Esta herramienta está optimizada para el análisis de variantes germinales en cohortes pequeñas y de variantes somáticas en pares de muestras tumorales-normales.

La ejecución de este programa se realiza en dos pasos: (1) configuración y (2) ejecución del flujo de trabajo. El paso de configuración se utiliza para especificar los datos de entrada y las opciones relacionadas con los métodos de llamada variantes, mientras que el paso de ejecución se usa para especificar cualquier parámetro relacionado con la forma en que se ejecuta.

Configuración:

```
{STRELKA_INSTALL_PATH}/bin/configureStrelkaGermlineWorkflow.py \
--bam file.bam \
--referenceFasta hg19.fa \
--runDir ${STRELKA_ANALYSIS_PATH}
```

Donde:

{STRELKA_INSTALL_PATH}/bin/configureStrelkaGermlineWorkflow.py es la ruta al archivo de configuración configureStrelkaGermlineWorkflow.py,

--bam file.bam especifica el archivo BAM de entrada,

--referenceFasta hg19.fa especifica el genoma de referencia que será utilizado para la llamada de variantes,

--runDir ${STRELKA_ANALYSIS_PATH} especifica la ruta donde se generará el archivo runWorkflow.py y los directorios de trabajo de strelka.

A continuación, se muestra como ejemplo el proceso de configuración para Strelka2 utilizado para una de las muestras del presente trabajo:

```
../../strelka-2.9.2.centos6_x86_64/bin/configureStrelkaGermlineWorkflow.py \
--bam 190830326_S6_paired_hg19_sorted.bam \
--referenceFasta hg19.fa \
--exome \
--runDir ../Analisis/strelka2/hg19/190830326_S6_paired_hg19_sorted
```

Ejecución:

En el paso de ejecución, se invoca el archivo runWorkflow.py generado durante el paso de configuración y a continuación se especifican los parámetros deseados:

```
{STRELKA_ANALYSIS_PATH}/runWorkflow.py
```

Donde {STRELKA_ANALYSIS_PATH}/runWorkflow.py especifica la ruta al archivo runWorkflow.py generado en el paso de configuración.

A continuación, se muestra como ejemplo el proceso de ejecución para Strelka2 utilizado para una de las muestras del presente trabajo:

```
../Analisis/strelka2/hg19/190830326_S6_paired_hg19_sorted/runWorkflow.py -m local
```

En este ejemplo, el parámetro –m local especifica que la ejecución se está llevando a cabo en una máquina normal. Si se quiere aumentar el número de núcleos para acelerar la ejecución, se podría hacer mediante el parámetro –j INT especificando el número de núcleos.

Tras la ejecución, se genera el archivo variants.vcf.gz, que contiene las variantes encontradas para la muestra analizada. Para evitar posteriores confusiones entre los archivos generados para cada paciente, para cada una de las muestras se cambia el nombre del archivo VCF mediante el siguiente comando:

```
mv variants.vcf.gz <filename>_strelka2.vcf
```

Como ejemplo para una de las muestras utilizadas:

```
mv variants.vcf.gz 190830326_S6_paired_hg19_sorted_strelka2.vcf
```

3.5.3. Picard

Construcción de archivo DICT

El segundo programa que se utilizará para llevar a cabo la llamada de variantes es Genome Analysis Toolkit (GATK)[28]. Para su ejecución, GATK requiere de la existencia previa de un archivo DICT en la misma carpeta que el archivo fasta correspondiente al genoma de referencia. Este archivo DICT es un diccionario de secuencia para una secuencia de referencia. El comando necesario para la creación de este archivo mediante Picard[29] es el siguiente Se muestra los comandos necesarios para la creación de los archivos DICT para hg19 y hg38:

```
java -jar picard.jar CreateSequenceDictionary R=hg19.fa O=hg19.dict

java -jar picard.jar CreateSequenceDictionary R=hg38.fa O=hg38.dict
```

donde:

java -jar picard.jar CreateSequenceDictionary invoca el programa,

R=hg19.fa indica el archivo FASTA que será procesado y

O= hg19.dict indica el archivo DICT que será generado.

Añadido de grupos de lectura:

Un segundo requisito para la llamada de variantes de GATK[28] es que los grupos de lectura estén especificados en el archivo BAM. Mediante la herramienta AddOrReplaceReadGroups de Picard[29], puede modificarse los archivos BAM añadiendo o remplazando estos grupos de lectura. Para ello se utiliza el siguiente comando (Se muestra como ejemplo el procesado de uno de los archivos BAM utilizados en el presente trabajo):

```
java -jar ../../picard.jar AddOrReplaceReadGroups \
I= 190830326_S6_paired_hg19_sorted.bam \
O= 190830326_S6_paired_hg19_sorted_RG.bam \
RGID=1 \
RGLB=lib1 \
RGPL=ILLUMINA \
RGPU=UNKNOWN \
RGSM=190830326_S6
```

Donde:

java -jar ../../picard.jar AddOrReplaceReadGroups invoca el programa,

I= 190830326_S6_paired_hg19_sorted.bam especifica el archivo BAM que será usado como input

O= 190830326_S6_paired_hg19_sorted_RG.bam es el archivo output que será generado.

RGID=1 Especifica el grupo de lectura,

RGLB=lib1 Especifica la librería,

RGPL=ILLUMINA especifica el tipo de plataforma,

RGPU=UNKNOWN especifica la unidad de plataforma,

RGSM=190830326_S6 especifica el nombre de la muestra.

3.5.4. Genome Analysis Toolkit (GATK)

Una vez generado el archivo DICT y añadidos los grupos de lectura, se puede llevar a cabo la llamada de variantes con gatk[28] utilizando su algoritmo HaplotypeCaller. Este algoritmo es capaz de detectar simultáneamente variaciones de nucleótido único e indels.

Para la ejecución de este programa se utiliza el siguiente comando (se utiliza como ejemplo el procesado de una de las muestras del presente estudio):

```
gatk HaplotypeCaller \
-R hg19.fa \
-I 190830326_S6_paired_hg19_sorted.bam \
-O 190830326_S6_paired_hg19_sorted.vcf
```

Donde:

gatk HaplotypeCaller invoca el algoritmo,

-R hg19.fa especifica el archivo FASTA que será utilizado como genoma de referencia

-I 190830326_S6_paired_hg19_sorted.bam especifica el archivo BAM utilizado como input,

-O 190830326_S6_paired_hg19_sorted.vcf especifica el archivo VCF que será generado.

3.6. Filtrado de archivos VCF

Durante la llamada de variantes se identifican miles de variantes, por lo que es conveniente realizar un filtrado antes de proseguir con el análisis, eliminando variantes que no pasan el filtro de calidad y restringiendo la búsqueda a las regiones de interés. Estas regiones son, en este caso, las regiones exónicas capturadas por la prueba y algunas regiones muy cercanas a exones que se capturan por azar o proximidad. Para ello se utiliza un archivo BED que contiene estas regiones de interés. Este archivo contiene las coordenadas de estas regiones para hg38, por lo que para filtrar los archivos VCF generados con hg38, se requiere de la transformación de estas coordenadas a hg19.

3.6.1. LiftOver

La transformación de coordenadas de hg38 a hg19 para el archivo BED que contenía las regiones de interés para el análisis se llevó a cabo con la herramienta liftOver. Para llevar a cabo esta transformación de coordenadas, es necesario un archivo CHAIN que contiene las

especificaciones para llevar a cabo esta transformación. La descarga de este archivo se llevó a cabo mediante el siguiente código:

```
wget http://hgdownload.soe.ucsc.edu/goldenPath/hg38/liftOver/hg38ToHg19.over.chain.gz
```

Para la ejecución del programa se utilizó el siguiente código:

```
./liftOver truseq-rapid-exome-probes-manifest-v1-2_hg38_expanded_+-50bp[9895].bed
hg38ToHg19.over.chain.gz truseq-rapid-exome-probes-manifest-v1-2_hg19_expanded_+-
50bp[9895].bed unlifted.bed
```

Donde ./liftOver invoca el algoritmo,

truseq-rapid-exome-probes-manifest-v1-2_hg38_expanded_+-50bp[9895].bed indica el archivo que será procesado,

hg38ToHg19.over.chain.gz es el archivo CHAIN que es utilizado para llevar a cabo la transformación de coordenadas,

truseq-rapid-exome-probes-manifest-v1-2_hg19_expanded_+-50bp[9895].bed es el archivo BED que será generado con las nuevas coordenadas,

unlifted.bed es el archivo BED que se genera con las coordenadas que no han podido ser transformadas.

3.6.2. BCFtools

El filtrado de variantes se llevó a cabo con BCFtools[26]. Esta herramienta permite la visualización y manipulación de archivos VCF y BCF, su variante binaria. Mientras que las variantes detectadas por GATK aparecen ya filtradas por calidad en el archivo VCF generado, strelka2 reporta todas las que detecta, pasen el filtro de calidad o no. Por tanto, el filtrado se ejecutó según regiones para los VCF generados por GATK y según regiones y calidad para los generados por strelka2. Para ello se utilizó el siguiente código (se utiliza como ejemplo el procesado de una de las muestras del presente estudio):

```
bcftools view -O z -o 190830326_S6_paired_hg19_sorted_strelka2_filtered.vcf.gz -f
PASS -R truseq-rapid-exome-probes-manifest-v1-2_hg19_expanded_+-50bp[9895].bed
190830326_S6_paired_hg19_sorted_strelka2.vcf.gz
```

Donde:

bcftools view invoca el algoritmo,

-O z especifica que el archivo generado sea un VCF comprimido,

-o 190830326_S6_paired_hg19_sorted_strelka2_filtered.vcf.gz especifica el archivo VCF comprimido que será generado,

-f PASS especifica el filtro para aquellas variantes que pasan el control de calidad,

-R truseq-rapid-exome-probes-manifest-v1-2_hg19_expanded_+-50bp[9895].bed especifica el archivo BED con las coordenadas de las regiones de interés,

190830326_S6_paired_hg19_sorted_strelka2.vcf.gz especifica el archivo que será procesado.

3.7. Recuento de variantes encontradas

3.7.1. BCFtools

El recuento de las variables encontradas se llevó a cabo con BCFtools[26]. Para ello, fue necesario el indexado previo de los archivos VCF. El código utilizado fue el siguiente:

```
bcftools index <file.vcf.gz>
```

Tras el indexado de los archivos, el número de variantes puede ser consultado mediante el siguiente comando:

```
Bcftools index --nrecords <file.vcf.gz>
```

Este recuento fue llevado a cabo sobre los archivos VCF antes y después de su filtrado.

3.8. Comparación de variantes encontradas en función del genoma de referencia

Uno de los principales objetivos de este trabajo es analizar las diferencias entre los análisis llevados a cabo con ambas versiones del genoma de referencia. Para ello, se debe comparar los archivos VCF generados con cada versión del genoma. Un requisito imprescindible para llevar a cabo esta comparación es que las coordenadas de ambos archivos a comparar sean compatibles, es decir, estén referenciadas a uno de los genomas de referencia elegidos.

3.8.1. Picard

Para llevar a cabo la conversión de coordenadas entre los archivos VCF se eligió la función LiftoverVcf de Picard[29]. Se generó el archivo convertido para todos los VCF generados durante la llamada de variantes. El código necesario para llevar a cabo la conversión es el siguiente:

```
java -jar picard.jar LiftoverVcf \ I=input.vcf \ O=lifted_over.vcf \ CHAIN=file.chain
\ REJECT=rejected_variants.vcf \ R=reference_sequence.fasta
```

Donde:

java –jar picard.jar LiftoverVcf invoca el algoritmo,

I=input.vcf especifica el archivo a ser convertido,

O=lifted_over.vcf especifica el archivo a ser generado,

CHAIN=file.chain especifica el archivo con las instrucciones necesarias para llevar a cabo la conversión de coordenadas

REJECT=rejected_variants.vcf especifica el archivo VCF donde se guardan las variantes para las que no se puede llevar a cabo la conversión entre versiones por incompatibilidades en las coordenadas.

R=reference_sequence.fasta especifica el archivo fasta con la referencia a la que será convertido

La obtención de los archivos CHAIN necesarios para la conversión de coordenadas entre hg38 y hg19 se llevó a cabo mediante el siguiente comando:

```
wget http://hgdownload.soe.ucsc.edu/goldenPath/hg38/liftOver/hg38ToHg19.over.chain.gz
wget http://hgdownload.soe.ucsc.edu/goldenPath/hg19/liftOver/hg19ToHg38.over.chain.gz
```

Como ejemplo, para una de los archivos incluidos en este estudio, el comando introducido fue el siguiente:

```
time java -jar -Xmx4G picard.jar LiftoverVcf \
I=190830326_S6_paired_hg19_sorted_RG_filtered_st.vcf \
O=190830326_S6_paired_hg19_sorted_RG_filtered_st_lifted_over_hg38.vcf \
CHAIN=hg19ToHg38.over.chain \
```

```
REJECT=190830326_S6_paired_hg19_sorted_RG_filtered_st_lifted_over_hg38_rejected_varia
nts.vcf \
R=hg38.fa
```

3.8.2. BCFtools

Una vez generados los archivos con las coordenadas transformadas entre versiones, la comparación entre ellos para cada una de los archivos VCF se llevó a cabo con la función isec de BCFtools[26]. Tras el indexado de los archivos VCF (ver sección 3.6.2) se aplicó la función isec. Esta función permite la obtención de las intersecciones entre dos archivos VCF, generando tras su comparación 4 archivos diferentes:

- 0000.vcf que contiene las variantes que solo se encuentran en el primero de los archivos especificado,
- 0001.vcf que contiene las variantes que solo se encuentran en el segundo de los archivos especificado,
- 0002.vcf que contiene las variantes que se encuentran en el primer archivo especificado, que también lo hacen en el segundo archivo especificado,
- 000.3vcf que contiene las variantes que se encuentran en el segundo archivo especificado, que también lo hacen en el primer archivo especificado.

El comando utilizado para la obtención de estos archivos es el siguiente:

```
bcftools isec –p folder file1.vcf.gz file2.vcf.gz
```

Donde:

bcftools isec invoca el algoritmo,

-p folder indica la carpeta donde se generarán los archivos

file1.vcf.gz especifica el primero de los archivos a comparar

file1.vcf.gz especifica el segundo de los archivos a comparar

Como ejemplo, para una de las comparaciones realizadas en el estudio, donde se comparaba las variantes encontradas con el uso del genoma de referencia hg19 en una de las muestras, en comparación con las variantes encontradas cuando el genoma de referencia era el hg38 el código utilizado fue el siguiente:

```
bcftools isec -p hg19_vs_hg38/190830326_S6_GATK \
190830326_S6_paired_hg19_sorted_RG_filtered_st.vcf.gz \
190830326_S6_paired_hg38_sorted_RG_filtered_st_lifted_over_hg19.vcf.gz
```

3.9. Anotación de archivos VCF

Hasta este momento, los archivos VCF generados contienen una información limitada y es conveniente añadir información adicional que permita una mejor interpretación de los resultados. Este paso se conoce como anotación. Durante la anotación de variantes, se agrega información referente a los efectos funcionales producidos por cada una de las variantes en el archivo VCF.

Existen diferentes herramientas anotadoras, como, por ejemplo: Variant Effect Predictor (VEP), Sort Intolerant From Tolerant (SIFT), Annotate Variation (ANNOVAR), o SnpEff. Aunque similares, los formatos en los que estas herramientas producen la anotación de variantes difieren en algunos aspectos y es conveniente conocer las características del escogido.

3.9.1. SnpEff

En nuestro caso, la herramienta escogida para llevar a cabo la anotación ha sido SnpEff, ya que es la utilizada por la mayoría de las principales instituciones de investigación y académicas, así como compañías farmacéuticas y proyectos de secuenciación clínica.

La información introducida durante la anotación es almacenada en el campo “ANN” del archivo VCF. En el manual online de SnpEff, concretamente en el apartado “ANN field (VCF Output Files)” se puede obtener información del campo “ANN”. La anotación con SnpEff se lleva a cabo con el siguiente comando:

```
java -jar -Xmx4G snpEff.jar GRCh37.75 file.vcf.gz -stats file_summary.html >
file.ann.vcf
```

Donde:

java -jar -Xmx4G snpEff.jar invoca el algoritmo,

GRCh37.75 especifica el genoma de referencia,

file.vcf.gz especifica el archivo que va a ser anotado,

-stats file_summary.html especifica el archivo HTML que será generado con la información estadística de las anotaciones.

file.ann.vcf especifica el archivo anotado que será generado.

3.10. Búsqueda de variantes patogénicas

Tras la anotación de los archivos VCF, se procedió a la búsqueda de las variantes patogénicas que habían sido previamente identificadas en los pacientes. Ya se disponía previamente de esta información, por lo que esta búsqueda se simplificó, realizándose manualmente mediante la visualización de los datos en la herramienta Integrative Genomics Viewer (IGV)[30], donde se accedió a la región genómica de interés y se realizó la búsqueda de la variante patogénica.

4. Resultados

4.1. Control de calidad

4.1.1. Datos brutos

Se analizó la calidad de los archivos FASTQ originales para comprobar su validez para posteriores análisis. El conjunto de informes generados fue procesado para su visualización conjunta con MultiQC.

Sample Name	**% Dups**	**% GC**	**M Seqs**
190830326_S6_R1_001	**40.8%**	**49%**	**70.5**
190830326_S6_R2_001	**39.4%**	**49%**	**70.5**
190831050_S7_R1_001	**46.9%**	**50%**	**68.7**
190831050_S7_R2_001	**45.6%**	**50%**	**68.7**
190837094_S7_R1_001	**41.5%**	**54%**	**58.8**
190837094_S7_R2_001	**39.8%**	**54%**	**58.8**
202713578_S1_R1_001	**35.9%**	**50%**	**39.7**
202713578_S1_R2_001	**34.9%**	**50%**	**39.7**

Tabla 2 Relación de archivos FASTQ procesados por FastQC (Datos Brutos)

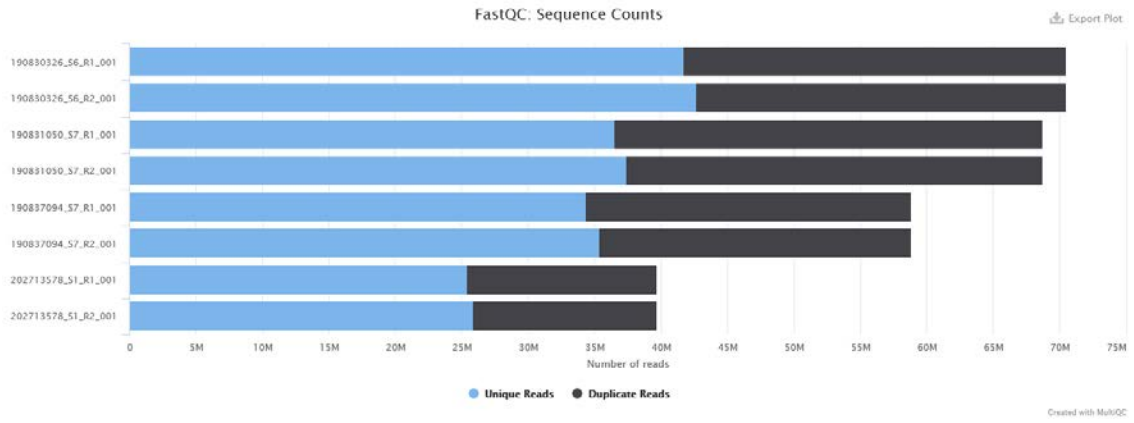
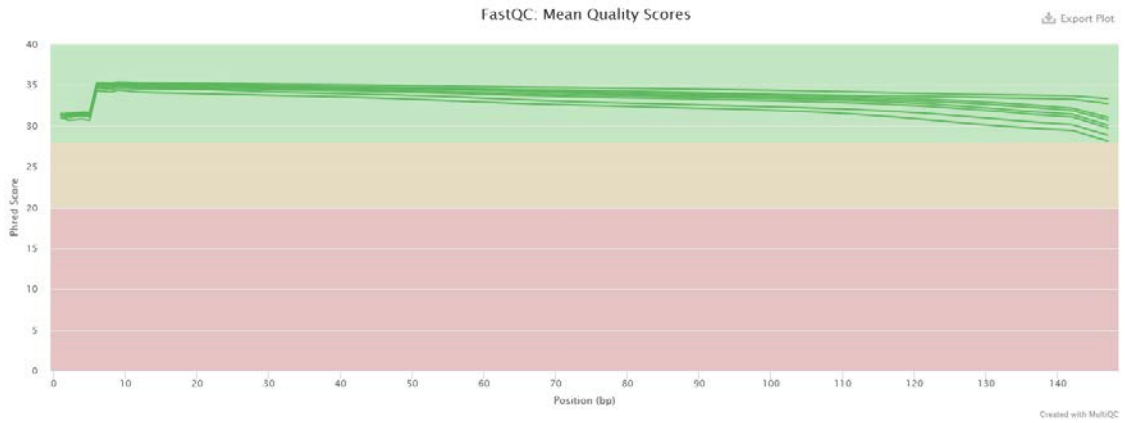


Figura 2 Longitud en millones de pares de bases de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Brutos)

a



b

Figura 3 Calidad de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Brutos)

Se midió la calidad de las lecturas contenidas en los archivos FASTQ (Phred Score), analizando la calidad media de cada base por su posición en la lectura (Figura 3a) y la calidad media a lo largo de todas las secuencias (Figura 3b). Se observó que, aunque tenían una calidad aceptable, una pequeña proporción, tenía Phred scores menores de lo deseado. Así mismo, se observó una menor calidad en las bases iniciales y finales de cada lectura en comparación con las bases centrales de las mismas.

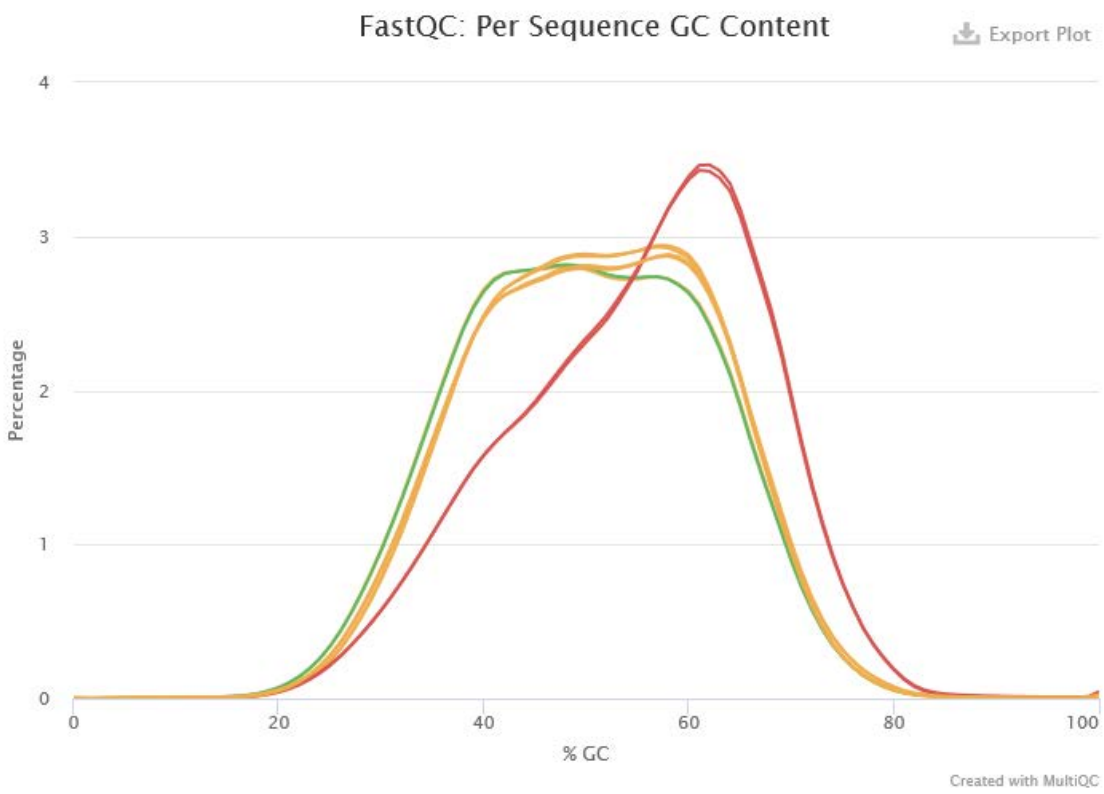


Figura 4 Contenido en CG de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Brutos)

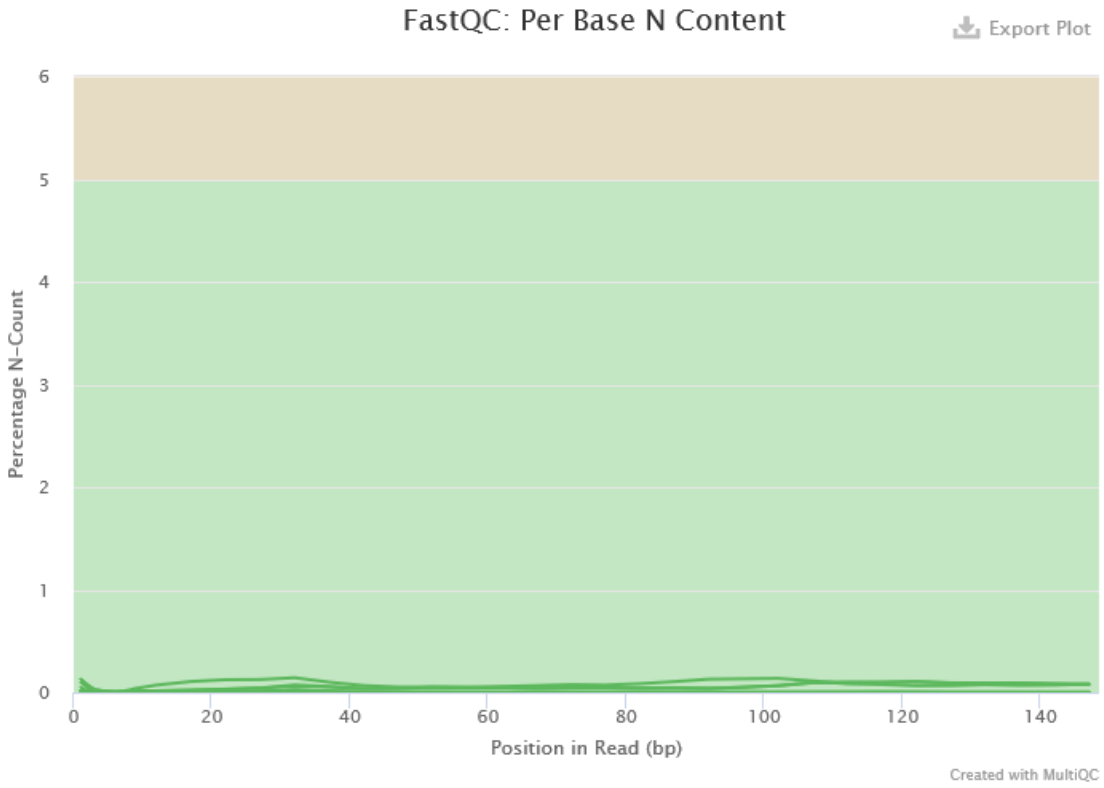


Figura 5 Contenido de bases indeterminadas (N) de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Brutos)

Se observó que el contenido en CG de las lecturas era variable, habiendo una distribución diferente en las diferentes muestras. Es destacable que una de las muestras difería significativamente respecto al resto en este aspecto (Figura 4).

Los niveles de bases indeterminadas detectados fueron bajos, no superando en ninguno de los casos el umbral que alertaría de una sobrerrepresentación de las mismas (Figura 4).

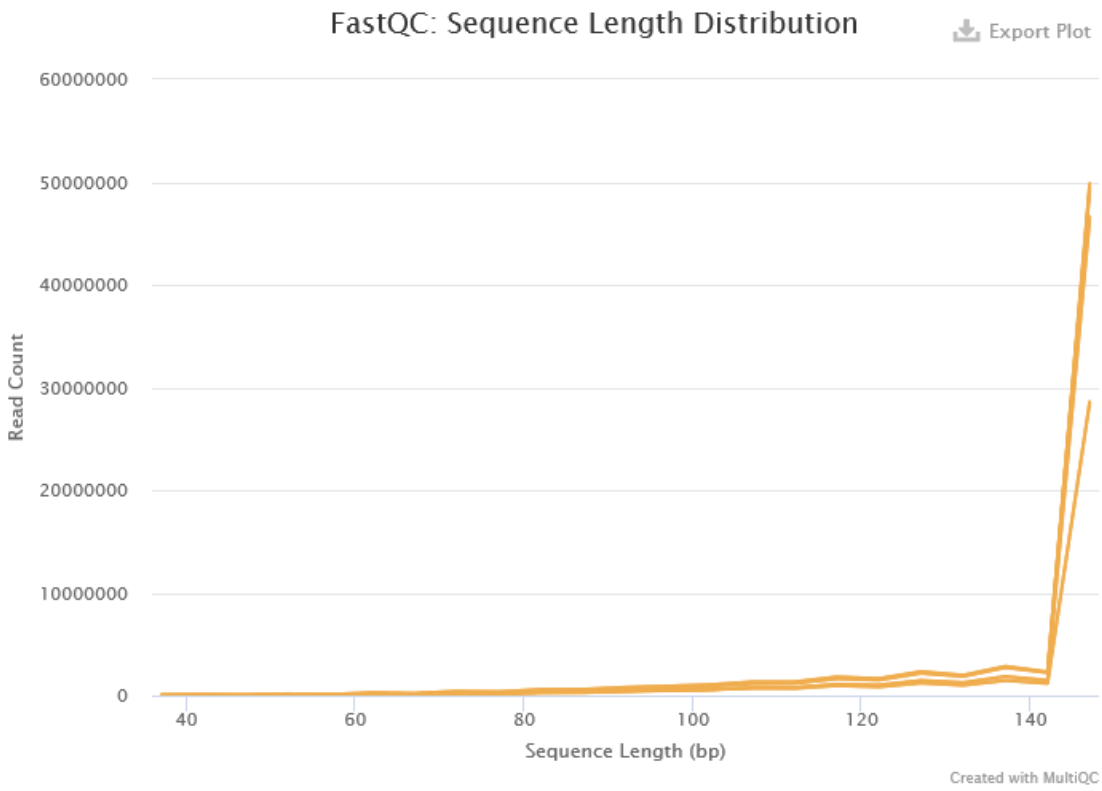


Figura 6 Distribución de tamaños de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Brutos)

Se detectó una longitud de lecturas suficientemente uniforme, habiendo una pequeña proporción de las mismas con una longitud menor de la deseada (Figura 6).

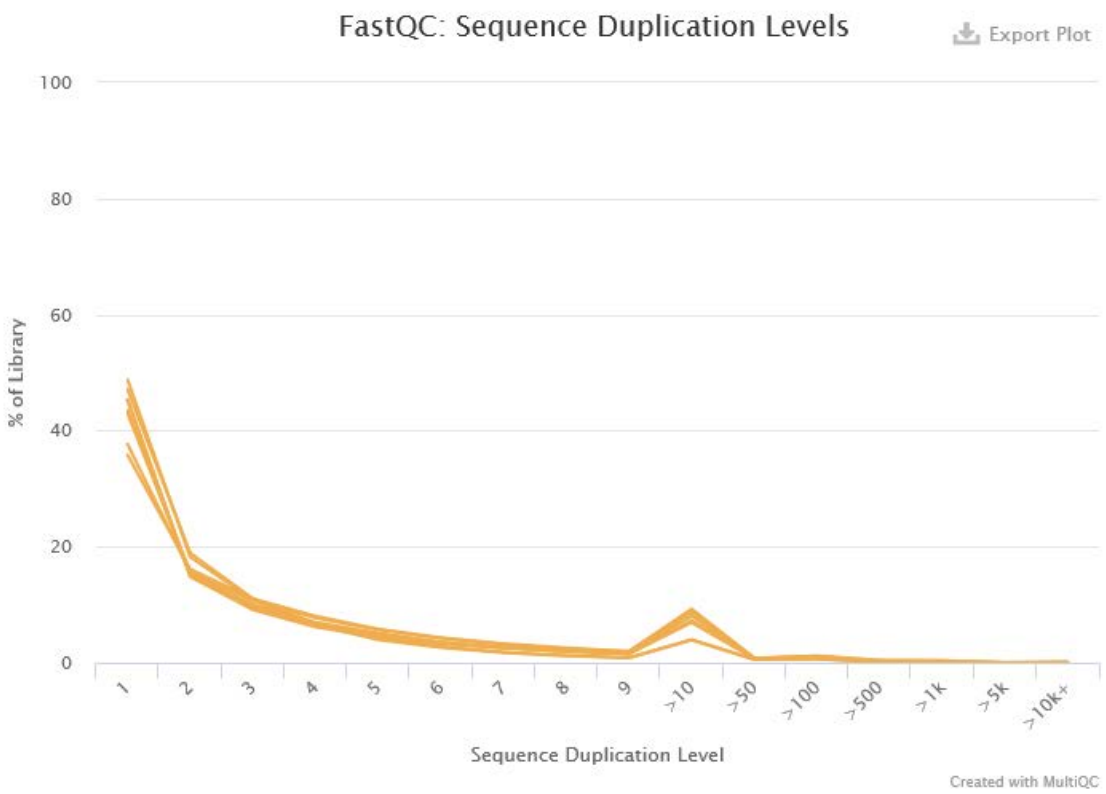


Figura 7 Nivel de duplicación de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Brutos)

Se detectaron niveles de duplicación superiores a los deseados (Figura 7).

4.1.2. Datos procesados

Tras su procesado, los archivos FASTQ generados fueron analizados con el fin de comprobar si el filtrado había mejorado la calidad de los mismos mediante FastQC. El conjunto de informes generados fue procesado para su visualización conjunta con MultiQC.

Sample Name	% Dups	% GC	M Seqs
190830326_S6_R1_001_paired	**40.1%**	**49%**	**51.6**
190830326_S6_R1_001_unpaired	**11.9%**	**50%**	**5.3**
190830326_S6_R2_001_paired	**40.1%**	**49%**	**51.6**
190830326_S6_R2_001_unpaired	**9.1%**	**51%**	**2.6**
190831050_S7_R1_001_paired	**46.0%**	**49%**	**52.5**
190831050_S7_R1_001_unpaired	**13.6%**	**51%**	**4.8**
190831050_S7_R2_001_paired	**46.4%**	**49%**	**52.5**
190831050_S7_R2_001_unpaired	**10.7%**	**52%**	**2.4**
190837094_S7_R1_001_paired	**40.2%**	**54%**	**37.4**
190837094_S7_R1_001_unpaired	**15.7%**	**54%**	**6.5**
190837094_S7_R2_001_paired	**40.4%**	**54%**	**37.4**
190837094_S7_R2_001_unpaired	**13.3%**	**55%**	**3.9**
202713578_S1_R1_001_paired	**35.4%**	**49%**	**31.1**
202713578_S1_R1_001_unpaired	**8.5%**	**51%**	**2.5**
202713578_S1_R2_001_paired	**35.4%**	**49%**	**31.1**
202713578_S1_R2_001_unpaired	**6.8%**	**52%**	**1.1**

Tabla 3 Relación de archivos FASTQ procesados por FastQC (Datos Procesados)

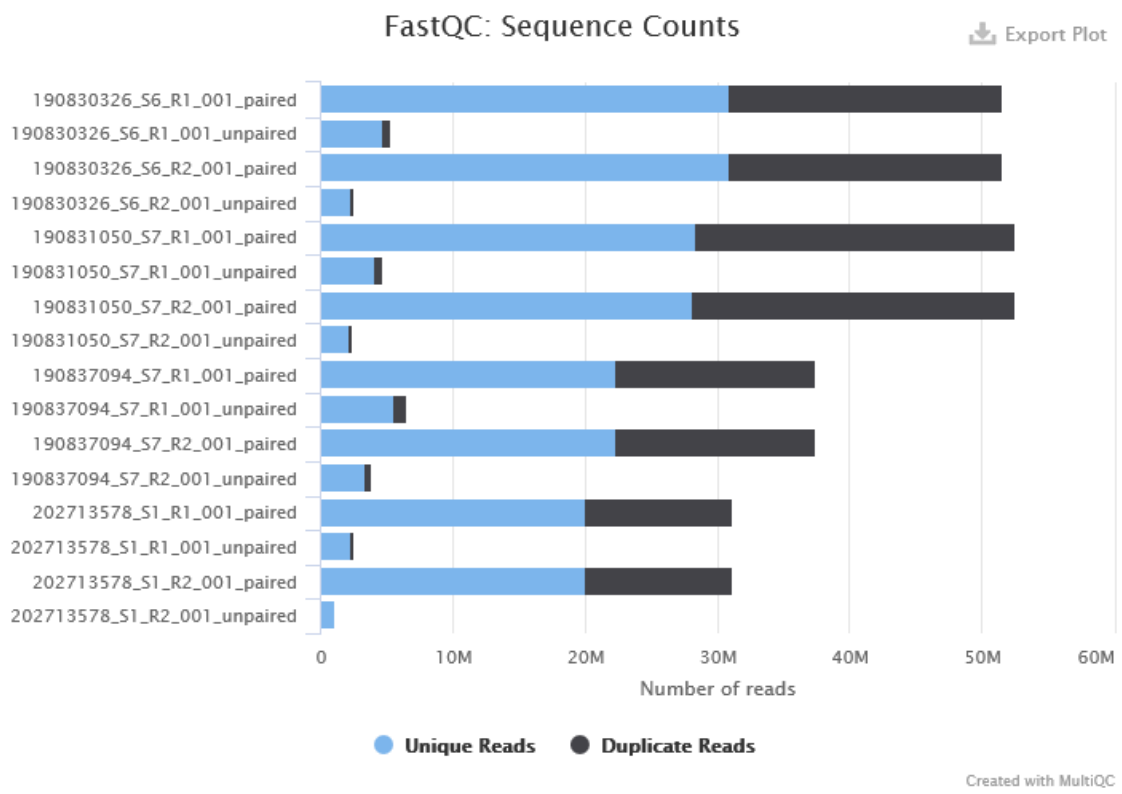


Figura 8 Longitud en millones de pares de bases de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Procesados)

Tras su procesado, la longitud de las lecturas contenidas en los archivos FASTQ se redujo significativamente (Figura 8 y Tabla 3).

a

b

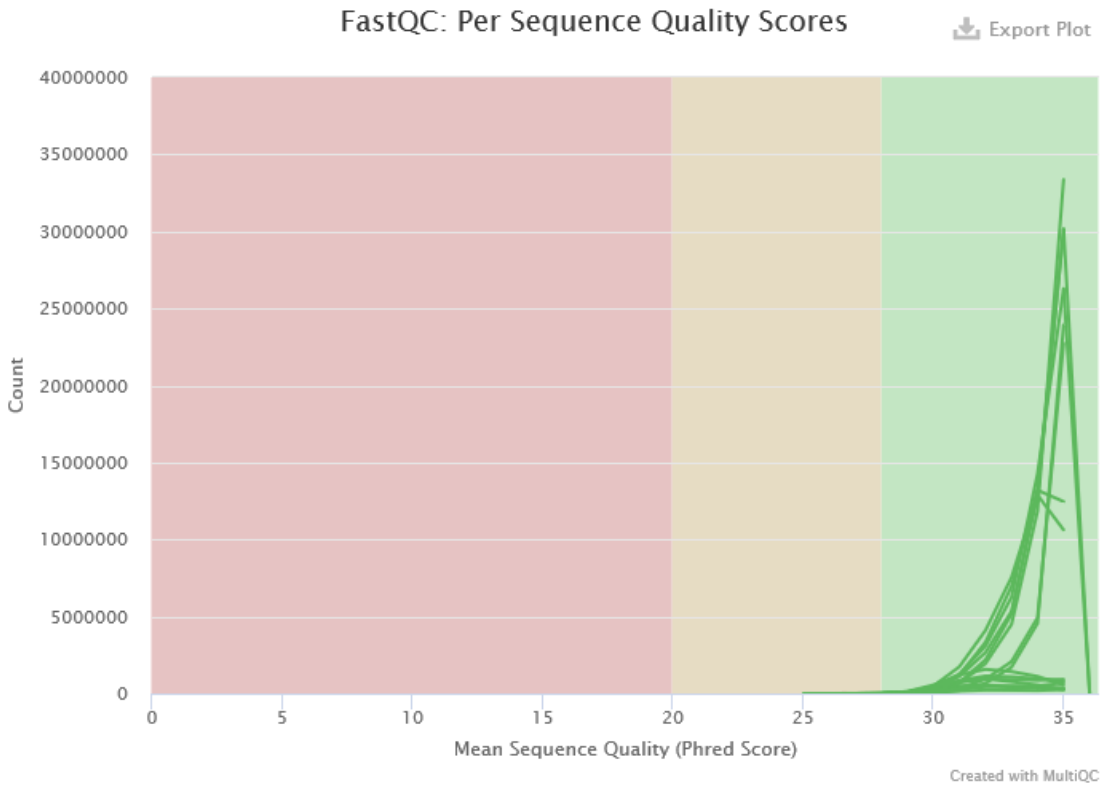


Figura 9 Calidad de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Procesados)

Se observó una mejora en la calidad de las bases finales de las lecturas (Figura 9a) y la calidad media de las lecturas contenidas en los archivos FASTQ procesados (Phred Score) mejoró significativamente (Figura 9b) con respecto a los archivos no procesados (Figura 3a).

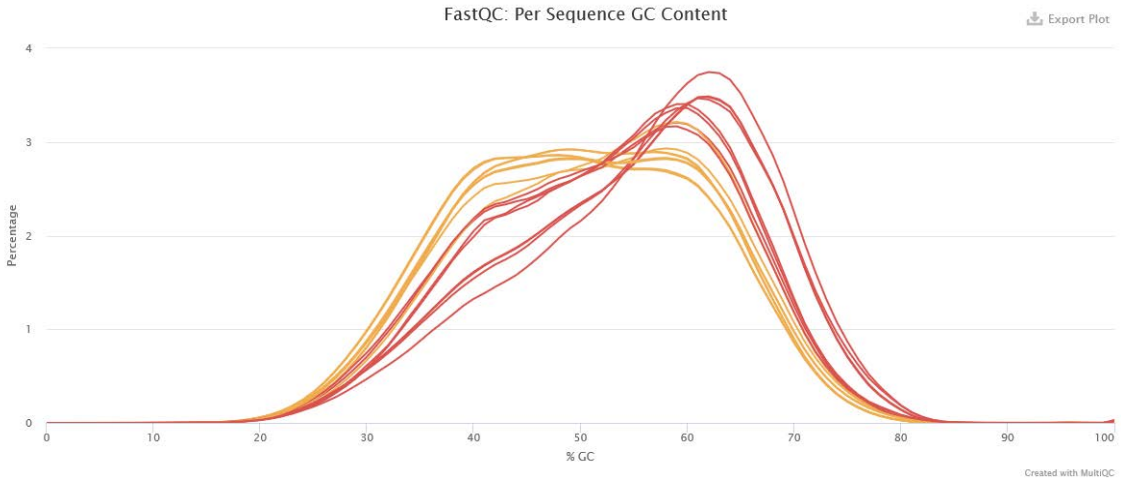


Figura 10 Contenido en CG de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Procesados)

Al igual que en las lecturas de los archivos no procesados (Figura 4), se observó que el contenido en CG de las lecturas era variable, habiendo una distribución diferente en las diferentes muestras (Figura 10).

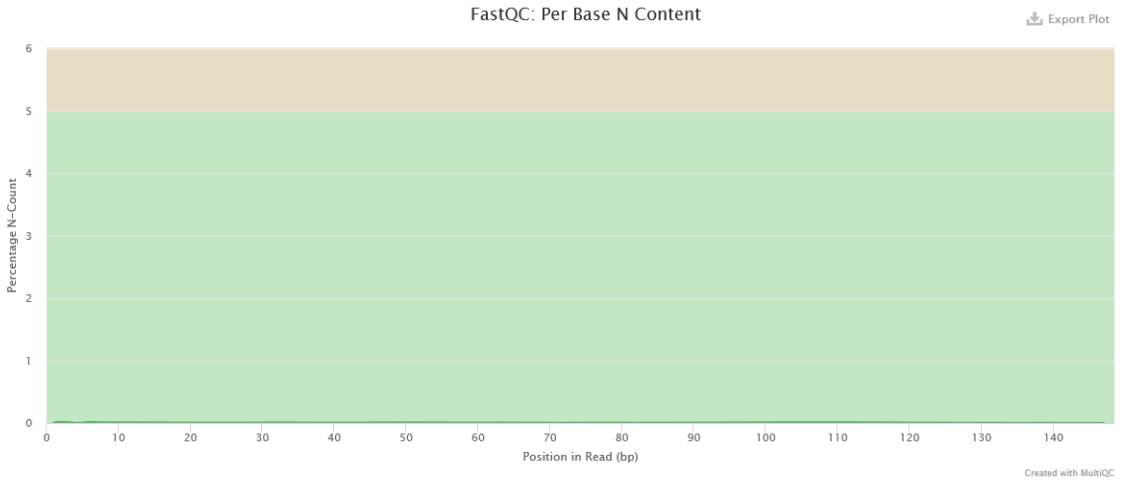


Figura 11 Contenido de bases indeterminadas (N) de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Procesados)

Se redujo el nivel de bases indeterminadas detectadas (Figura 11) con respecto al observado en los archivos no procesados (Figura 4).

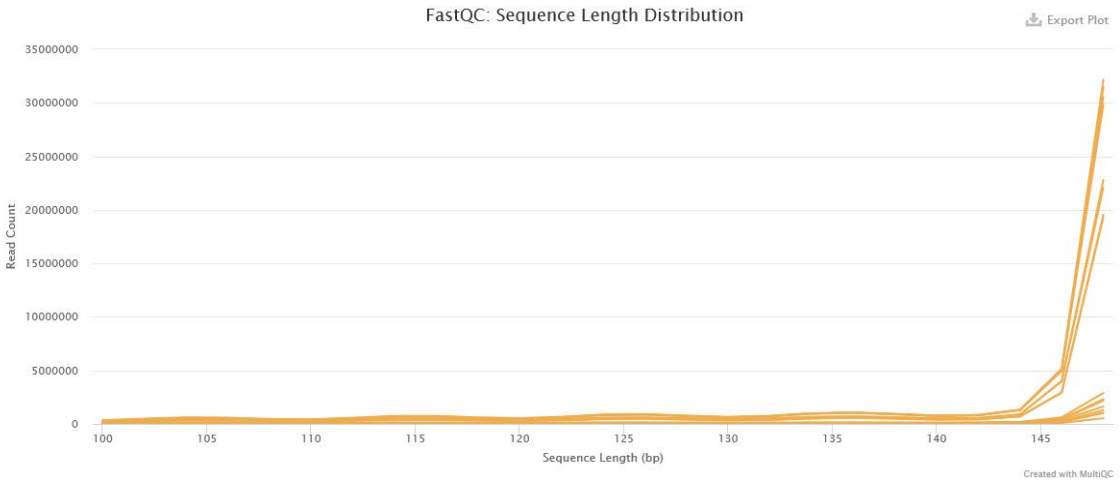


Figura 12 Distribución de tamaños de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Procesados)

La uniformidad en la longitud de las lecturas fue mejorada, filtrando todas aquellas lecturas con una longitud menor de 100 pares de bases (Figura 12).

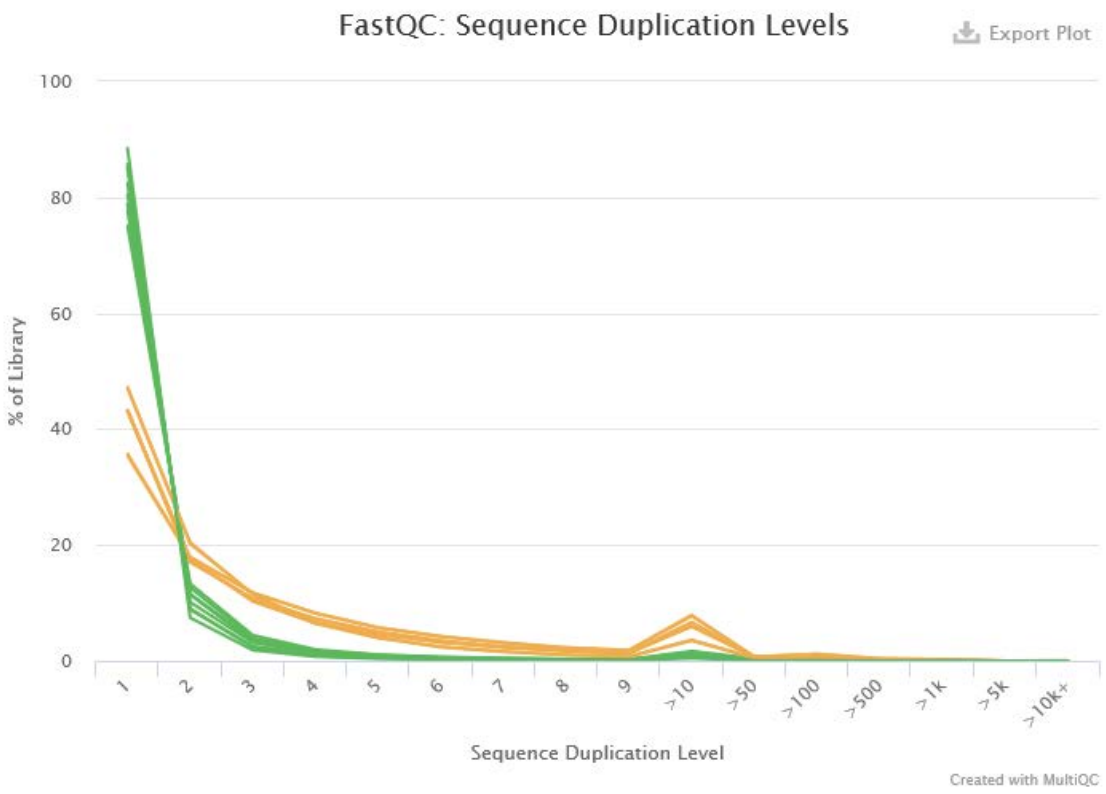


Figura 13 Nivel de duplicación de las lecturas contenidas en los archivos FASTQ procesados por FastQC (Datos Procesados)

El nivel de duplicación de secuencias siguió siendo mayor del óptimo tras el procesado de los archivos FASTQ.

4.1. Relación de datos filtrados

Se contabilizó la cantidad de secuencias contenidas, su porcentaje de duplicación, su porcentaje de CG y el número de pares de bases de los archivos FASTQ filtrados y se comparó con el de los archivos sin filtrar (Tabla 4).

Sample Name	% Dups	% GC	M Seqs	M Filtered Paired (%)	M Filtered Unpaired (%)	M Discard (%)	% Dups (Filtered Paired)	% Dups (Filtered Unpaired)	% GC (Filtered Paired)	% GC (Filtered Unpaired)
190830326_S6_R1_001	40,80%	49%	70,5	51,6 (73,2%)	5,3 (7,5%)	13,6 (19,3%)	40.1%	11.9%	49%	50%
190830326_S6_R2_001	39,40%	49%	70,5	51,6 (73,2%)	2,6 (3,7%)	16,3 (23,1%)	40.1%	9.1%	49%	51%
190831050_S7_R1_001	46,90%	50%	68,7	52,5 (76,4%)	4,8 (7,0%)	11,4 (16,6%)	46.0%	13.6%	49%	51%
190831050_S7_R2_001	45,60%	50%	68,7	52,5 (76,4%)	2,4 (3,5%)	13,8 (20,1%)	46.4%	10.7%	49%	52%
190837094_S7_R1_001	41,50%	54%	58,8	37,4 (63,6%)	6,5 (11,1%)	14,9 (25,3%)	40.2%	15.7%	54%	54%
190837094_S7_R2_001	39,80%	54%	58,8	37,4 (63,6%)	3,9 (6,6%)	17,5 (29,8%)	40.4%	13.3%	54%	55%
202713578_S1_R1_001	35,90%	50%	39,7	31,1 (78,3%)	2,5 (6,3%)	6,1 (15,4%)	35.4%	8.5%	49%	51%
202713578_S1_R2_001	34,90%	50%	39,7	31,1 (78,3%)	1,1 (2,8%)	7,5 (18,9%)	35.4%	6.8%	49%	52%

Tabla 4 Relación de datos filtrados

4.2. Relación de variantes encontradas

Se contabilizó el número de variantes encontradas por cada uno de los programas utilizados y se contabilizó las diferencias debidas a la elección de uno u otro genoma de referencia (Tabla 5).

NO FILTRADAS

muestra		strelka2			GATK		
		hg19	hg38	%DIFF	hg19	hg38	%DIFF
	190830326_S6	1655997	1572757	5,0	1121666	1056328	5,8
	190831050_S7	1045559	992208	5,1	702958	658521	6,3
	190837094_S7	1088634	1035147	4,9	611712	571649	6,6
	202713578_S1	822980	780931	5,1	524197	490369	6,5

Tabla 5 Relación de variantes encontradas (No filtradas)

Tras el filtrado de las variantes por regiones de interés, se volvió a contabilizar el número de variantes (Tabla 6).

FILTRADAS

muestra		Número de variantes					
		strelka2			GATK		
		hg19	hg38	%DIFF	hg19	hg38	%DIFF
	190830326_S6	61637	58233	5,5	66457	61789	7,0
	190831050_S7	60084	56769	5,5	64330	59917	6,9
	190837094_S7	60118	56669	5,7	66036	61349	7,1
	202713578_S1	59741	56585	5,3	64177	59945	6,6

Tabla 6 Relación de variantes encontradas (Filtradas)

Se calculó el porcentaje de variantes que habían sido filtradas por regiones de interés y calidad de las lecturas en cada uno de los casos (Tabla 7).

% FILTRADAS

muestra		strelka2		GATK	
		hg19	hg38	hg19	hg38
	190830326_S6	3,7	3,7	5,9	5,8
	190831050_S7	5,7	5,7	9,2	9,1
	190837094_S7	5,5	5,5	10,8	10,7
	202713578_S1	7,3	7,2	12,2	12,2

Tabla 7 Porcentaje de variantes que pasan el filtro

4.3. Identificación de variantes patogénicas

Dado que se disponía del diagnóstico genético de cada uno de los pacientes incluidos en el estudio, se procedió a la búsqueda de las mutaciones previamente identificadas en cada uno de ellos mediante la visualización de las variantes anotadas con el software Integrative Genomics Viewer (IGV). En todos los casos, estas variantes patogénicas habían sido identificadas

correctamente, independientemente del software utilizado para la búsqueda de variantes y del genoma de referencia. En la siguiente tabla se puede consultar las variantes patogénicas que, en cualquiera de los casos fue identificada correctamente.

202713578_S1			
GUSB	c.1084G>C	p.Asp362His	Homozigot
190837094_S7			
SLC6A8	c.1681G	p.Gly561Arg	Hemizigot
190831050_S7			
OTC	c.622G>A	p.Ala208Thr	Hemizigot
190830326_S6			
ACAT1	c.1189C>A	p.His397Asn	homozigot

Tabla 8 Mutación patogénica encontrada en cada paciente

4.4. Análisis de las diferencias en detección de variantes

Tras anotar los archivos VCF correspondientes a las variantes cuya detección difería con el uso de uno u otro genoma de referencia, se procedió a la clasificación de estas variantes atendiendo a diferentes criterios como el tipo de variante, su posible efecto, su clase funcional y la región génica afectada. En las siguientes tablas se muestra el número de variantes encontradas con el uso de un genoma de referencia y no con el otro. Las comparaciones hg19vshg38 hacen referencia a variantes que son detectadas con el uso del genoma de referencia hg19, pero no se detectan cuando se utiliza el hg38 y las comparaciones hg38vshg19 hacen referencia a variantes que son detectadas con el uso del genoma de referencia hg38, pero no se detectan cuando se utiliza el hg19.

4.4.1. Clasificación por tipo de variante:

muestra	comparación	variant caller	SNP	INS	DEL
190830326_S6	hg19vshg38	GATK	5304	297	303
190831050_S7	hg19vshg38	GATK	4986	292	302
190837094_S7	hg19vshg38	GATK	5356	302	296
202713578_S1	hg19vshg38	GATK	4841	291	282
190830326_S6	hg19vshg38	Strelka2	3935	255	211
190831050_S7	hg19vshg38	Strelka2	3768	258	219
190837094_S7	hg19vshg38	Strelka2	3950	270	214
202713578_S1	hg19vshg38	Strelka2	3679	249	198
190830326_S6	hg38vshg19	GATK	1115	43	68
190831050_S7	hg38vshg19	GATK	1047	37	73
190837094_S7	hg38vshg19	GATK	1157	38	66
202713578_S1	hg38vshg19	GATK	1073	42	59
190830326_S6	hg38vshg19	Strelka2	907	37	47
190831050_S7	hg38vshg19	Strelka2	835	29	54
190837094_S7	hg38vshg19	Strelka2	898	33	53
202713578_S1	hg38vshg19	Strelka2	891	30	44

Tabla 9 Diferencias en detección de variantes por tipo

4.4.2. Clasificación por estimación del efecto.

muestra	comparación	variant caller	HIGH	LOW	MODERATE	MODIFIER
190830326_S6	hg19vshg38	GATK	956	3684	3741	29375
190831050_S7	hg19vshg38	GATK	1068	3636	3543	28933
190837094_S7	hg19vshg38	GATK	1003	3668	3782	30269
202713578_S1	hg19vshg38	GATK	780	3367	3434	27779
190830326_S6	hg19vshg38	Strelka2	803	2992	2978	23616
190831050_S7	hg19vshg38	Strelka2	952	3092	2840	23832
190837094_S7	hg19vshg38	Strelka2	764	3051	2902	24327
202713578_S1	hg19vshg38	Strelka2	610	2936	2765	22758
190830326_S6	hg38vshg19	GATK	16	44	23	4419
190831050_S7	hg38vshg19	GATK	13	31	17	4184
190837094_S7	hg38vshg19	GATK	10	45	30	4467
202713578_S1	hg38vshg19	GATK	16	41	24	4243
190830326_S6	hg38vshg19	Strelka2	20	42	23	3650
190831050_S7	hg38vshg19	Strelka2	15	33	17	3390
190837094_S7	hg38vshg19	Strelka2	12	36	22	3414
202713578_S1	hg38vshg19	Strelka2	17	39	26	3441

Tabla 10 Diferencias en detección de variantes por efecto

4.4.3. Clasificación por clase funcional.

muestra	comparación	variant caller	MISSENSE	NONSENSE	SILENT
190830326_S6	hg19vshg38	GATK	3588	67	2908
190831050_S7	hg19vshg38	GATK	3457	80	2783
190837094_S7	hg19vshg38	GATK	3649	85	2745
202713578_S1	hg19vshg38	GATK	3328	66	2484
190830326_S6	hg19vshg38	Strelka2	2888	53	2378
190831050_S7	hg19vshg38	Strelka2	2767	52	2349
190837094_S7	hg19vshg38	Strelka2	2798	65	2285
202713578_S1	hg19vshg38	Strelka2	2681	53	2205
190830326_S6	hg38vshg19	GATK	21	3	11
190831050_S7	hg38vshg19	GATK	15	3	3
190837094_S7	hg38vshg19	GATK	28	3	12
202713578_S1	hg38vshg19	GATK	24	3	3
190830326_S6	hg38vshg19	Strelka2	23	3	13
190831050_S7	hg38vshg19	Strelka2	17	3	5
190837094_S7	hg38vshg19	Strelka2	22	3	14
202713578_S1	hg38vshg19	Strelka2	26	3	3

Tabla 11 Diferencias en detección de variantes por clase funcional

4.4.4. Clasificación por región génica afectada.

muestra	*comparación*	*variant caller*	*INTER-GENIC*	*UP-STREAM*	*UTR 5 PRIME*	*SPLICE SITE ACCEPTOR*	*SPLICE SITE DONOR*	*SPLICE SITE REGION*	*EXON*	*INTRON*	*UTR 3 PRIME*	*DOWN-STREAM*
190830326_S6	hg19vshg38	GATK	217	5631	531	93	84	686	10882	11679	953	6984
190831050_S7	hg19vshg38	GATK	209	5418	559	94	53	757	10643	11346	949	7142
190837094_S7	hg19vshg38	GATK	196	5975	546	99	70	799	10943	12040	1017	7018
202713578_S1	hg19vshg38	GATK	207	5294	557	90	66	726	9911	10942	932	6627
190830326_S6	hg19vshg38	Strelka2	146	4478	406	87	45	543	8595	9540	805	5729
190831050_S7	hg19vshg38	Strelka2	160	4405	439	81	40	667	8572	9455	821	6070
190837094_S7	hg19vshg38	Strelka2	148	4758	429	77	62	665	8441	9822	860	5767
202713578_S1	hg19vshg38	Strelka2	133	4214	432	82	52	624	8033	9144	849	5501
190830326_S6	hg38vshg19	GATK	634	998	9	2	6	12	144	1932	88	673
190831050_S7	hg38vshg19	GATK	580	987	5	0	2	7	125	1813	73	652
190837094_S7	hg38vshg19	GATK	675	984	5	2	2	7	159	1872	94	750
202713578_S1	hg38vshg19	GATK	475	737	5	2	2	7	157	1695	74	578
190830326_S6	hg38vshg19	Strelka2	475	737	5	2	2	7	157	1695	74	578
190831050_S7	hg38vshg19	Strelka2	412	732	3	2	0	7	119	1583	60	536
190837094_S7	hg38vshg19	Strelka2	493	613	5	2	2	7	132	1571	73	583
202713578_S1	hg38vshg19	Strelka2	457	672	9	2	2	16	140	1510	67	644

Tabla 12 Diferencias en detección de variantes por región génica

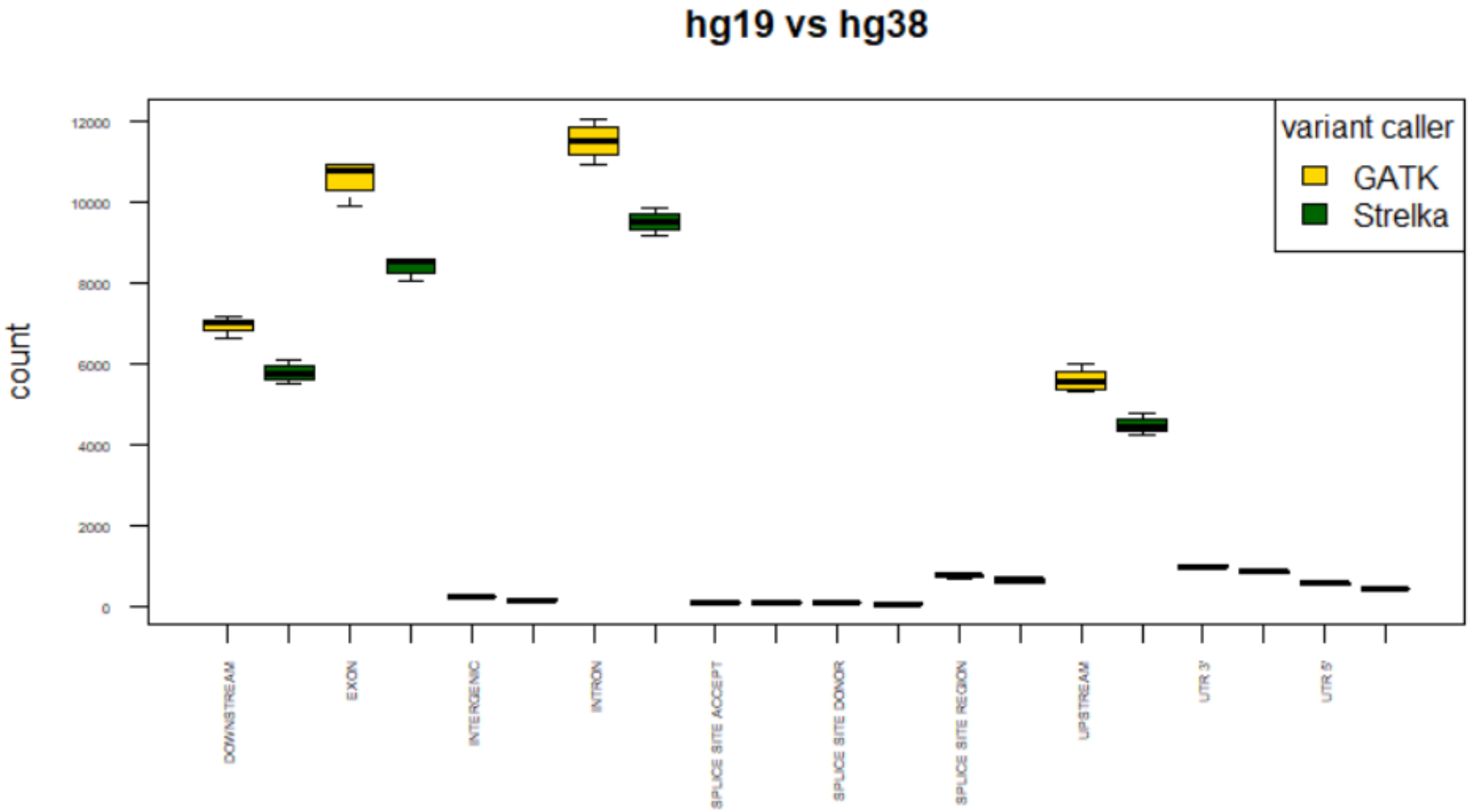


Figura 14 Diferencias en detección de variantes por región génica. Variantes que se detectan con hg19 y no con hg38

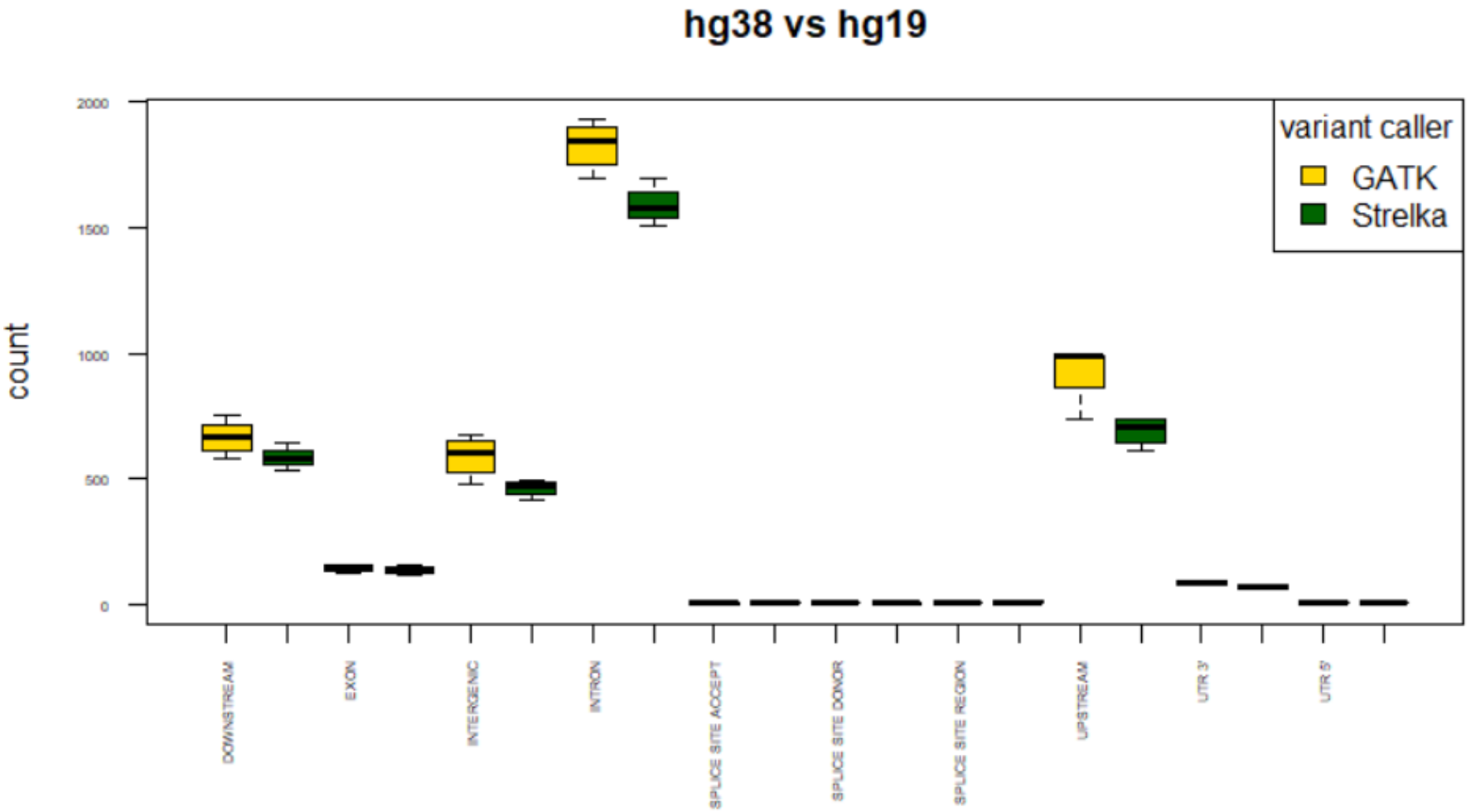


Figura 15 Diferencias en detección de variantes por región génica. Variantes que se detectan con hg38 y no con hg19

5. Discusión

La realización de este estudio pretende aportar una respuesta simple a la pregunta ¿Qué genoma de referencia es mejor utilizar durante la llamada de variantes? Para responder a esta pregunta es necesario discernir entre dos elementos de la misma. En primer lugar, hay que analizar qué diferencias existen entre el uso de uno u otro genoma de referencia en cuanto a las variantes detectadas y, en segundo lugar, se debe averiguar cuál es la posible relevancia clínica de aquellas variantes cuya detección difiere con el uso de diferentes genomas de referencia.

A raíz de este estudio, se ha observado cómo el número de variantes detectadas es mayor cuando se utiliza el genoma de referencia hg19 que cuando se utiliza el hg38 (Tabla 6), no existiendo una gran diferencia en la calidad de las lecturas utilizadas para la detección de dichas variantes (Tabla 7). Esto contradice los resultados de estudios previos, que mostraban una detección de un mayor número de variantes con el uso de hg38 como genoma de referencia[11]. Una posible explicación para este hecho es la inclusión de haplotipos alternativos en la versión hg38 del genoma. Durante el desarrollo de esta nueva versión, se produjo un esfuerzo a la hora de incorporar al genoma de referencia, poblaciones más distantes de las utilizadas en la anterior versión. Esto tiene implicaciones muy reales, incluso en términos de resultados clínicos, ya que la capacidad para identificar variaciones significativas en la secuencia del genoma de un individuo depende directamente de la posibilidad de distinguir variantes normales de patológicas. Además, esta nueva versión del genoma, corrige miles de pequeños artefactos de secuenciación que provocan falsos positivos en la detección de SNP e indels. Por tanto, en el momento actual, es totalmente recomendable la elección de hg38 como genoma de referencia de elección para realizar la llamada de variantes.

Los diferentes softwares utilizan diferentes algoritmos para la detección de estas variantes, por lo que siempre es recomendable el uso de más de uno de ellos para evitar la pérdida de información. En el caso de este estudio, el uso de GATK ha reportado un mayor número de variantes que Strelka.

Asumiendo por tanto una mayor fiabilidad de los resultados obtenidos con el uso de hg38 como genoma de referencia, cobra una mayor importancia el análisis del potencial efecto patológico de aquellas variantes que se detectan con el uso de este genoma de referencia, mientras que no lo hacen con hg19. Dados los medios disponibles, el enfoque que se ha seguido en el presente estudio ha sido el de la clasificación de estas variantes atendiendo a diferentes criterios como el tipo de variante, su posible efecto, su clase funcional y la región génica afectada.

Atendiendo al tipo de variante, la inmensa mayoría de variantes encontradas son polimorfismos de nucleótido simple, mientras que solo un pequeño porcentaje serían variantes del tipo indel (Tabla 9). En cuanto al impacto en la clínica de este tipo de variantes, se ha reportado que el impacto de ambos tipos de variación en enfermedades complejas no es diferente[31,32]. En todos los casos, el número de variantes detectados de cada tipo fue mayor con el uso de GATK que con Strelka (*paired-t.test-p-value* < 0.05).

Se ha realizado una segunda clasificación atendiendo al posible efecto sobre el individuo producido por la variante y asignando este efecto a 4 categorías predefinidas según su importancia. Para mayor detalle en esta clasificación, puede consultarse el manual de snpEff en (http://snpeff.sourceforge.net/SnpEff_manual.html), *Impact prediction*. Aunque en este aspecto, la mayoría de las variantes encontradas tienen un efecto modificador-moderado, hay un pequeño número de estas para las que se predice un gran efecto (Tabla 10). El número de variantes detectadas por GATK con un potencial gran efecto fue significativamente mayor que las detectadas por strelka (*paired-t.test-p-value* = 0.03739). Un análisis en detalle de estas variantes se escapa a las aspiraciones de este estudio, pero deja la puerta abierta a la continuación del mismo.

Atendiendo a la clase funcional de aquellas variantes que afectan a zonas codificantes, la gran mayoría son mutaciones de cambio de sentido, mientras que el menor número lo representan las variantes de pérdida de sentido. En cuanto a variantes sinónimas, el número de estas varía entre las diferentes muestras incluidas en el estudio (Tabla 11). En este caso, no hubo diferencias significativas en el número de variantes detectadas de cada tipo con el uso de los diferentes softwares.

Por último, atendiendo a la localización por regiones de las áreas afectadas por estas variantes, se aprecia como la mayoría de ellas se localizan en zonas no codificantes, mientras que solo unas pocas de ellas aparecen en regiones exónicas (Tabla 12). En cuanto a las diferencias encontradas según el software utilizado, GATK encontró un numero significativamente mayor (*paired-t.test-p-value* < 0.05) de variantes que Strelka en intrones, zonas intergénicas, zonas *upstream* y extremos UTR 3'. Puede llevar a error pensar que aquellas variantes situadas en zonas no codificantes no son capaces de generar efectos adversos, pues estas pueden estar afectando a zonas reguladoras que controlen la expresión final de las proteínas, entendiendo como tal todas aquellas implicadas en los procesos como la unión de ARN de interferencia, la alteración de los patrones de metilación del ADN, la alteración de la estructura local de ARNs no codificantes o la regulación génica a través de enhancers y promotores entre otros[13,14] .

Como se ha podido observar, el uso de genomas de referencia como herramientas para la búsqueda de variantes tiene sus limitaciones, y la constante actualización de estos genomas con nueva información es necesaria para mejorar este tipo de análisis conforme el conocimiento avanza. Además, existen diferencias genéticas entre diferentes poblaciones que necesitan ser tenidas en cuenta a la hora de decidir si una determinada variante es normal o no en una población. Por tanto, la tendencia actual está llevando a la sustitución de estos genomas de referencia por los denominados genomas consenso, que son diferentes para cada grupo poblacional y por tanto recogen mejor la variabilidad interindividual[33].

6. Conclusiones

1. Se ha llevado a cabo la llamada de variantes a partir de datos de NGS utilizando los genomas de referencia hg19 y hg38, habiendo detectado un mayor número de variantes con el uso del genoma de referencia hg19.
2. Tras analizar las diferencias entre los análisis llevados a cabo con ambas versiones del genoma, se ha encontrado un alto número de variantes detectadas por cada uno de los genomas de referencia que no se detectan cuando se utiliza el otro.
3. Existe un pequeño número de variantes con posibles efectos deletéreos cuya detección no sería posible con el uso de hg19 como genoma de referencia, por lo que se recomienda el uso de hg38 como genoma de referencia para la llamada de variantes con propósitos clínicos.

7. Glosario

ADN	Ácido desoxiribonucleico
ARN	Ácido ribonucléico
BAM	"Binary alignment map"
BED	"Browser extensible data"
BWA	"Burrows-wheeler aligner "
CG	Citosina y guanina
FASTA	Formato de fichero informático basado en texto, utilizado para representar secuencias usando códigos de una única letra
FASTQ	Archivos de texto que contienen datos de secuencia con una puntuación de calidad (Phred) para cada base, representada como un carácter ASCII
GATK	"Genome analysis toolkit"
GRCh37	"Genome Research Consortium human build 37"
GRCh38	"Genome Research Consortium human build 38"
HG19	"Human genome build 19"
HG38	"Human genome build 38"
HTML	Lenguaje de marcas de hipertexto ("hypertext Markup Language")
IGV	"Integrative genomics viewer"
indels	Inserciones y deleciones
NGS	Secuenciación de próxima generación (“next generation sequencing”)
SAM	"Sequence alignment map"
SNV	Variantes de nucleótido único (“sinle nucleotide variant”)
UCSC	Universidad de California en Santa Cruz
VCF	"Variant call format"

8. Bibliografía

1. Venter JC, Adams MD, Myers EW, et al. The sequence of the human genome. *Science*. 2001;291(5507):1304-1351.

2. Lander ES, Linton LM, Birren B, et al. Initial sequencing and analysis of the human genome. *Nature*. 2001;409(6822):860-921.

3. Genome reference consortium. https://www.ncbi.nlm.nih.gov/grc/help/faq/#human-reference-genome-individuals. Updated 2020. Accessed 06/11, 2020.

4. Shastry BS. SNPs in disease gene mapping, medicinal drug development and evolution. *J Hum Genet*. 2007;52(11):871-880.

5. Davey JW, Hohenlohe PA, Etter PD, Boone JQ, Catchen JM, Blaxter ML. Genome-wide genetic marker discovery and genotyping using next-generation sequencing. *Nat Rev Genet*. 2011;12(7):499-510.

6. Linderman MD, Brandt T, Edelmann L, et al. Analytical validation of whole exome and whole genome sequencing for clinical applications. *BMC Med Genomics*. 2014;7:20-8794-7-20.

7. Muzzey D, Evans EA, Lieber C. Understanding the basics of NGS: From mechanism to variant calling. *Curr Genet Med Rep*. 2015;3(4):158-165.

8. Ashley EA. Towards precision medicine. *Nat Rev Genet*. 2016;17(9):507-522.

9. Ye H, Meehan J, Tong W, Hong H. Alignment of short reads: A crucial step for application of next-generation sequencing data in precision medicine. *Pharmaceutics*. 2015;7(4):523-541.

10. Schneider VA, Graves-Lindsay T, Howe K, et al. Evaluation of GRCh38 and de novo haploid genome assemblies demonstrates the enduring quality of the reference assembly. *Genome Res*. 2017;27(5):849-864.

11. Pan B, Kusko R, Xiao W, et al. Similarities and differences between variants called with human reference genome HG19 or HG38. *BMC Bioinformatics*. 2019;20(Suppl 2):101-019-2620-0.

12. Pan B, Kusko R, Xiao W, et al. Similarities and differences between variants called with human reference genome HG19 or HG38. *BMC Bioinformatics*. 2019;20(Suppl 2):101-019-2620-0.

13. Katsonis P, Koire A, Wilson SJ, et al. Single nucleotide variations: Biological impact and theoretical interpretation. *Protein Sci*. 2014;23(12):1650-1666.

14. Ramirez-Bello J, Vargas-Alarcon G, Tovilla-Zarate C, Fragoso JM. Single nucleotide polymorphisms (SNPs): Functional implications of regulatory-SNP (rSNP) and structural RNA (srSNPs) in complex diseases. *Gac Med Mex*. 2013;149(2):220-228.

15. Yohe S, Thyagarajan B. Review of clinical next-generation sequencing. *Arch Pathol Lab Med*. 2017;141(11):1544-1557.

16. Meyer M, Kircher M. Illumina sequencing library preparation for highly multiplexed target capture and sequencing. *Cold Spring Harb Protoc*. 2010;2010(6):pdb.prot5448.

17. Maróti Z, Boldogkői Z, Tombácz D, Snyder M, Kalmár T. Evaluation of whole exome sequencing as an alternative to BeadChip and whole genome sequencing in human population genetic analysis. *BMC Genomics*. 2018;19(1):778-018-5168-x.

18. Ulintz PJ, Wu W, Gates CM. Bioinformatics analysis of whole exome sequencing data. *Methods Mol Biol*. 2019;1881:277-318.

19. LaDuca H, Farwell KD, Vuong H, et al. Exome sequencing covers >98% of mutations identified on targeted next generation sequencing panels. *PLoS One*. 2017;12(2):e0170843.

20. Nakagawa H, Fujita M. Whole genome sequencing analysis for cancer genomics and precision medicine. *Cancer Sci*. 2018;109(3):513-522.

21. Shen T, Pajaro-Van de Stadt SH, Yeat NC, Lin JC. Clinical applications of next generation sequencing in cancer: From panels, to exomes, to genomes. *Front Genet*. 2015;6:215.

22. FastQC: A quality control tool for high throughput sequence data [online]. . 2015.

23. Ewels P, Magnusson M, Lundin S, KÃ¤ller M. MultiQC: Summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*. 2016;32(19):3047-3048.

24. Bolger, A. M., Lohse, M., & Usadel, B. Trimmomatic: A flexible trimmer for illumina sequence data. *Bioinformatics*. 2014;btu170.

25. Li H, Durbin R. Fast and accurate short read alignment with burrows-wheeler transform. *Bioinformatics*. 2009;25(14):1754-1760.

26. Li H, Handsaker B, Wysoker A, et al. The sequence alignment/map format and SAMtools. *Bioinformatics*. 2009;25(16):2078-2079.

27. Kim S, Scheffler K, Halpern AL, et al. Strelka2: Fast and accurate calling of germline and somatic variants. *Nat Methods*. 2018;15(8):591-594.

28. McKenna A, Hanna M, Banks E, et al. The genome analysis toolkit: A MapReduce framework for analyzing next-generation DNA sequencing data. *Genome Res*. 2010;20(9):1297-1303.

29. Picard toolkit. *Broad Institute, GitHub repository*. 2018.

30. Robinson JT, Thorvaldsdóttir H, Winckler W, et al. Integrative genomics viewer. *Nat Biotechnol*. 2011;29(1):24-26.

31. Gagliano SA, Sengupta S, Sidore C, et al. Relative impact of indels versus SNPs on complex disease. *Genet Epidemiol*. 2019;43(1):112-117.

32. Montgomery SB, Goode DL, Kvikstad E, et al. The origin, evolution, and functional impact of short insertion-deletion variants identified in 179 human genomes. *Genome Res*. 2013;23(5):749-761.

33. Ballouz S, Dobin A, Gillis JA. Is it time to change the reference genome? *Genome Biol*. 2019;20(1):159-019-1774-4.

FastQC. Módulos de análisis.

Basic Statistics:

Este módulo ofrece características descriptivas del archivo analizado:

- **Filename**: El nombre del archivo original analizado.
- **File type**: Indica si el archivo contiene las bases reales o los datos de imagen preprocesados a partir de los cuales se obtienen estas bases.
- **Encoding**: Indica qué codificación ASCII de valores de calidad se encontró en este archivo.
- **Total Sequences**: Un recuento del número total de secuencias procesadas.
- **Filtered Sequences**: Informa del número de secuencias eliminadas cuando el programa se ejecuta en modo Casava, que se utiliza para el análisis de archivos casava FASTQ, un tipo especial de archivo FASTQque contiene datos de una misma muestra separados en varios archivos que contienen secuencias de baja calidad que se han marcado para ser eliminadas.
- **Sequence Length**: Proporciona la longitud de la secuencia más corta y más larga del conjunto. Si todas las secuencias tienen la misma longitud, solo se informa un valor.
- **% GC**: El% GC general de todas las bases en todas las secuencias.

Per Base Sequence Quality:

Muestra una descripción general del rango de valores de calidad en todas las bases en cada posición en el archivo FASTQ. Gráficamente, se muestra un diagrama de cajas para cada una de las posiciones de cada lectura, con la mediana y rango intercuartílico representados por la línea roja y caja amarilla respectivamente y los rangos 10% y 90% representados por los bigotes, así como una línea azul que representa la calidad media.

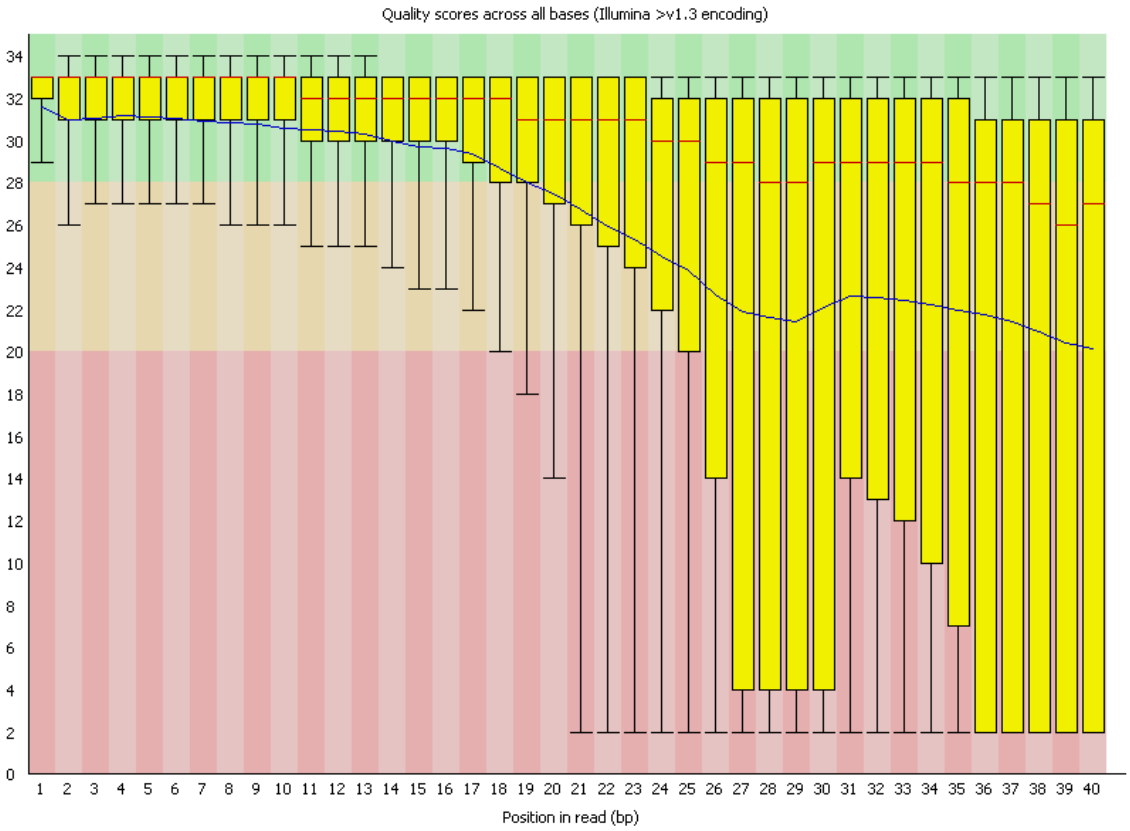


Figura 16 Per Base Sequence Quality (Ejemplo)

Este módulo considera que la muestra no supera el mínimo de calidad cuando el cuartil inferior para cualquier base es menor que 5 o si la mediana para cualquier base es menor que 20 y muestra una advertencia si el cuartil inferior para cualquier base es inferior a 10, o si la mediana para cualquier base es inferior a 25.

Por lo general, la calidad de las lecturas va empeorando a medida que se avanza en la longitud de la secuencia a causa de la degradación de los adaptadores utilizados. La detección de un fallo en este módulo hace recomendable un procesado de los datos con el fin de eliminar las secuencias cuya calidad sea menor del umbral deseado.

Per Sequence Quality Scores:

Muestra gráficamente la puntuación de calidad por secuencia, permitiendo observar la presencia de subconjuntos de secuencias con valores de calidad universalmente bajos.

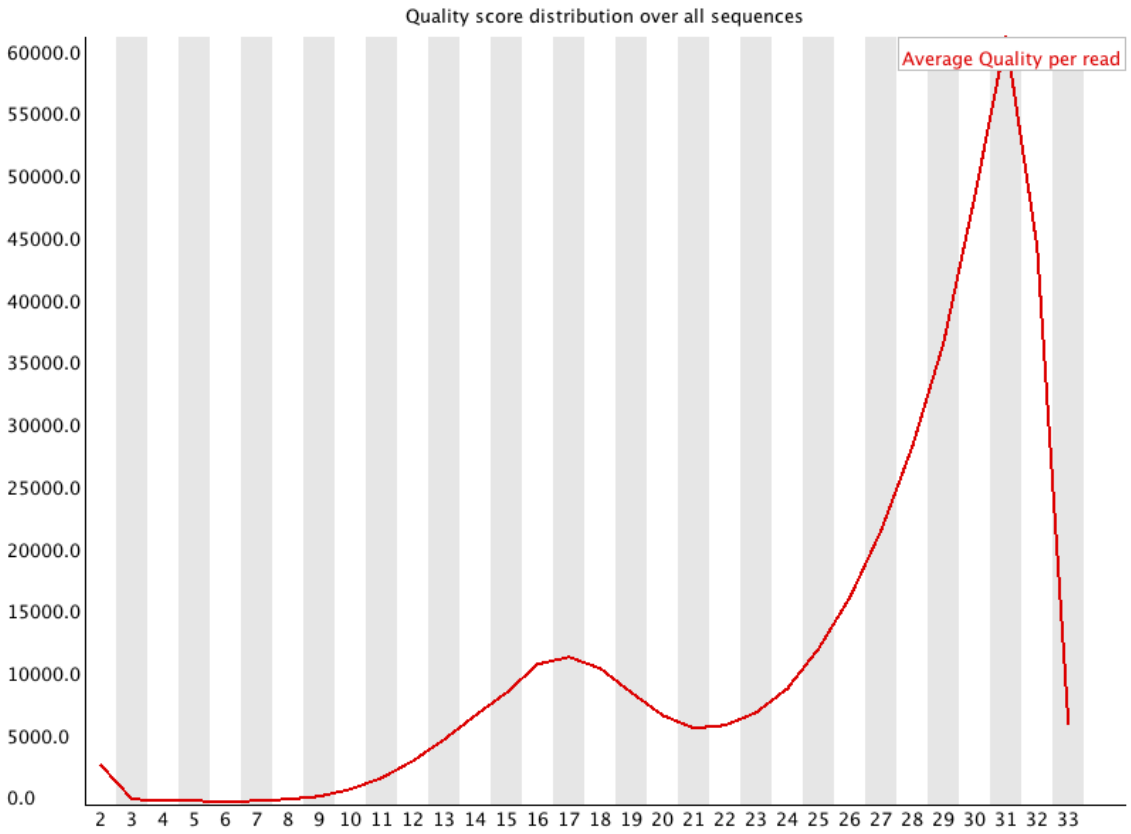


Figura 17 Per Sequence Quality Scores (Ejemplo)

Este módulo genera un error si la calidad media observada con más frecuencia es inferior a 20, lo que equivale a una tasa de error del 1% y muestra una advertencia si la calidad media observada con más frecuencia es inferior a 27, lo que equivale a una tasa de error del 0.2%.

Per Base Sequence Content:

Muestra gráficamente la proporción que cada base representa en cada posición de las lecturas.

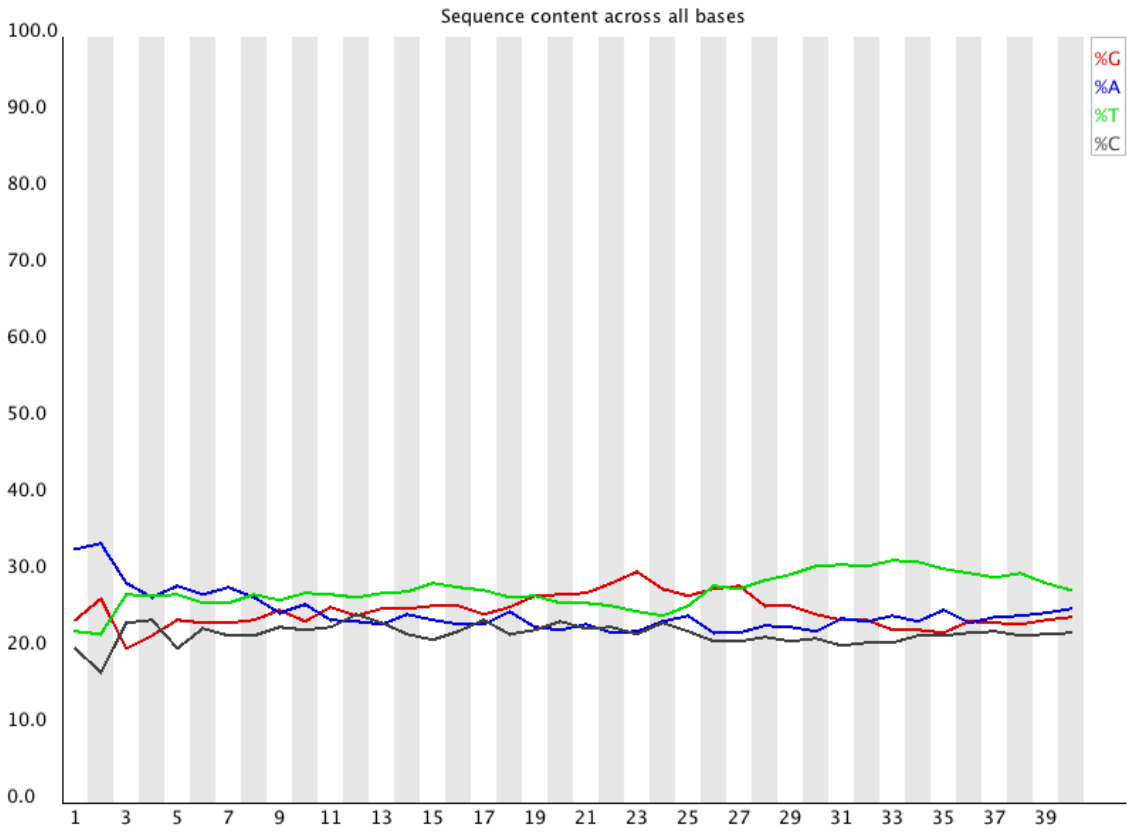


Figura 18 Per Base Sequence Content (Ejemplo)

En una biblioteca aleatoria, se esperaría que hubiera poca o ninguna diferencia entre las diferentes bases de una ejecución de secuencia, por lo que las líneas en este gráfico deben correr paralelas entre sí. La cantidad relativa de cada base debe reflejar la cantidad total de estas bases en su genoma, pero en cualquier caso no deben estar muy desequilibradas entre sí.

Algunos tipos de biblioteca siempre producen una composición de secuencia sesgada, normalmente en el extremo 5 'de las lecturas a causa de una selección sesgada inespecífica de los cebadores, aunque este sesgo no parece afectar negativamente el análisis posterior.

Este módulo fallará si la diferencia entre A y T, o G y C es mayor al 20% en cualquier posición y dará una advertencia si la diferencia entre A y T, o G y C es superior al 10% en cualquier posición.

Per Sequence GC Content:

Este módulo mide el contenido de GC en toda la longitud de cada secuencia en un archivo y lo compara con una distribución normal modelada de contenido de GC.

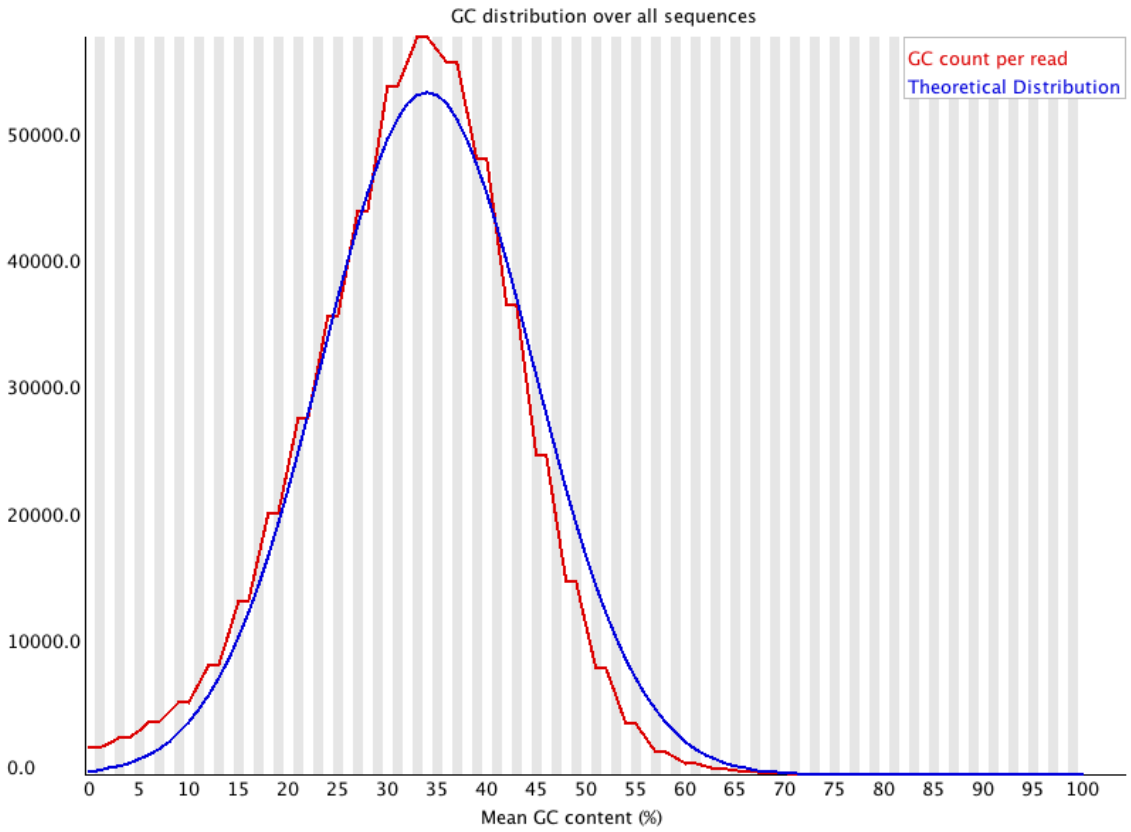


Figura 19 Per Sequence GC Content: (Ejemplo)

En una librería aleatoria normal, se esperaría ver una distribución más o menos normal del contenido de GC donde el pico central corresponde al contenido general de GC del genoma de la especie a la que pertenece.

Este módulo fallará si la suma de las desviaciones de la distribución normal representa más del 30% de las lecturas y mostrará una advertencia si la suma de las desviaciones de la distribución normal representa más del 15% de las lecturas.

Per Base N Content:

Este módulo muestra para cada posición el porcentaje de bases codificadas como N. Esta codificación es la que se da por los secuenciadores cuando no se puede realizar una llamada base con suficiente confianza.

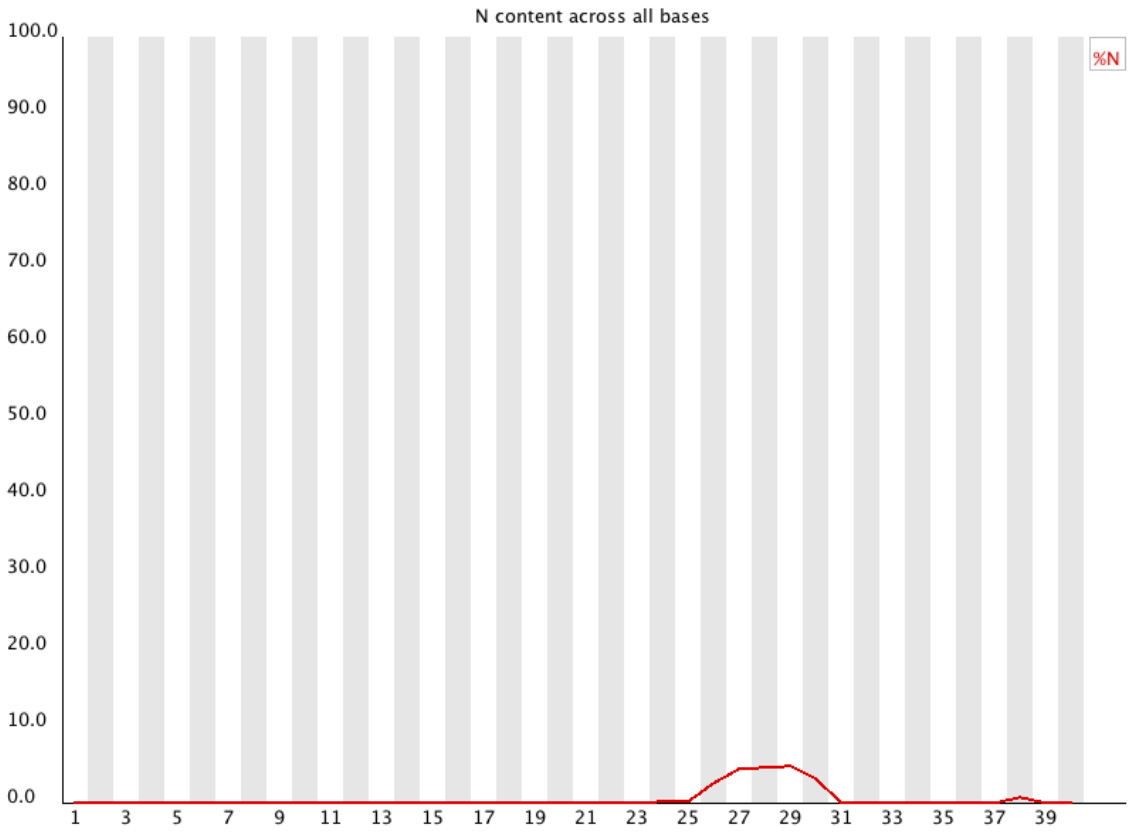


Figura 20 Per Base N Content (Ejemplo)

Este módulo generará un error si alguna posición muestra un contenido de N> 20% y mostrará una advertencia si cualquier posición muestra un contenido de N> 5%.

Sequence Length Distribution

Genera un gráfico que muestra la distribución de los tamaños de fragmentos en el archivo que se analizó.

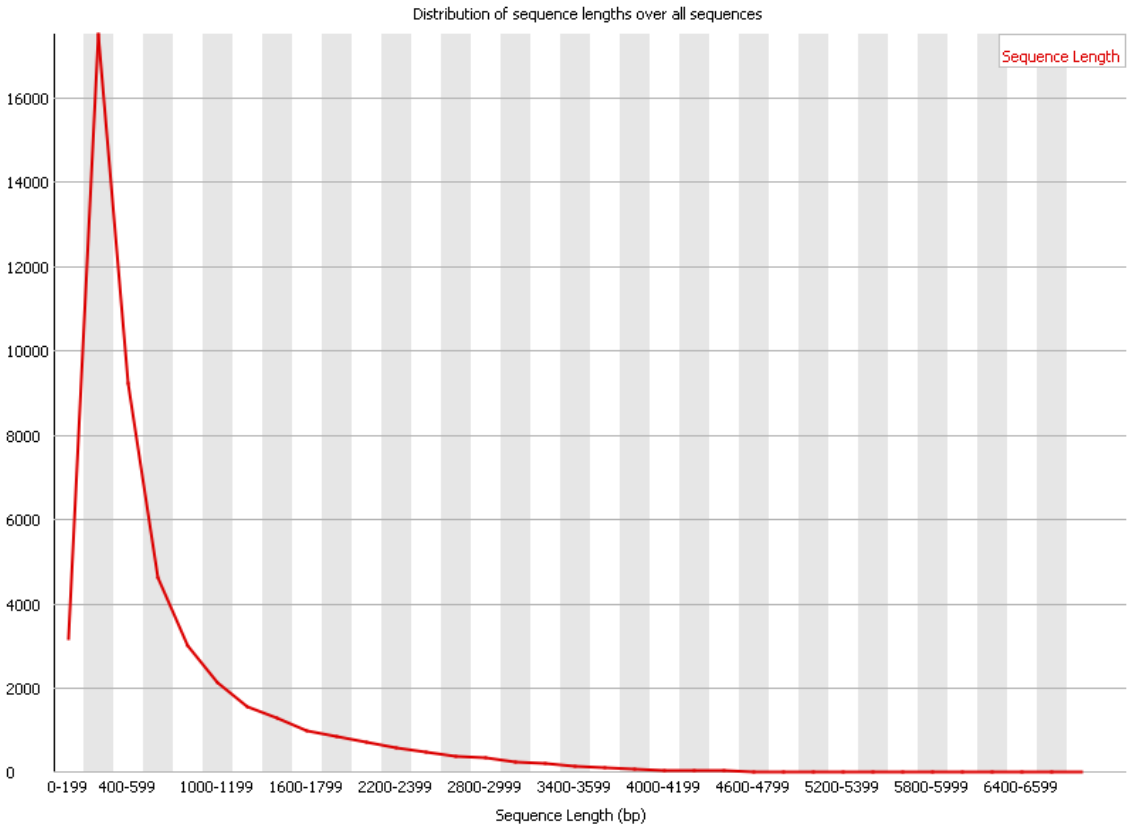


Figura 21 Sequence Length Distribution (Ejemplo)

Este módulo generará un error si alguna de las secuencias tiene longitud cero o una advertencia si todas las secuencias no tienen la misma longitud.

Duplicate sequences:

Este módulo cuenta el grado de duplicación para cada secuencia en una librería y crea un gráfico que muestra el número relativo de secuencias con diferentes grados de duplicación.

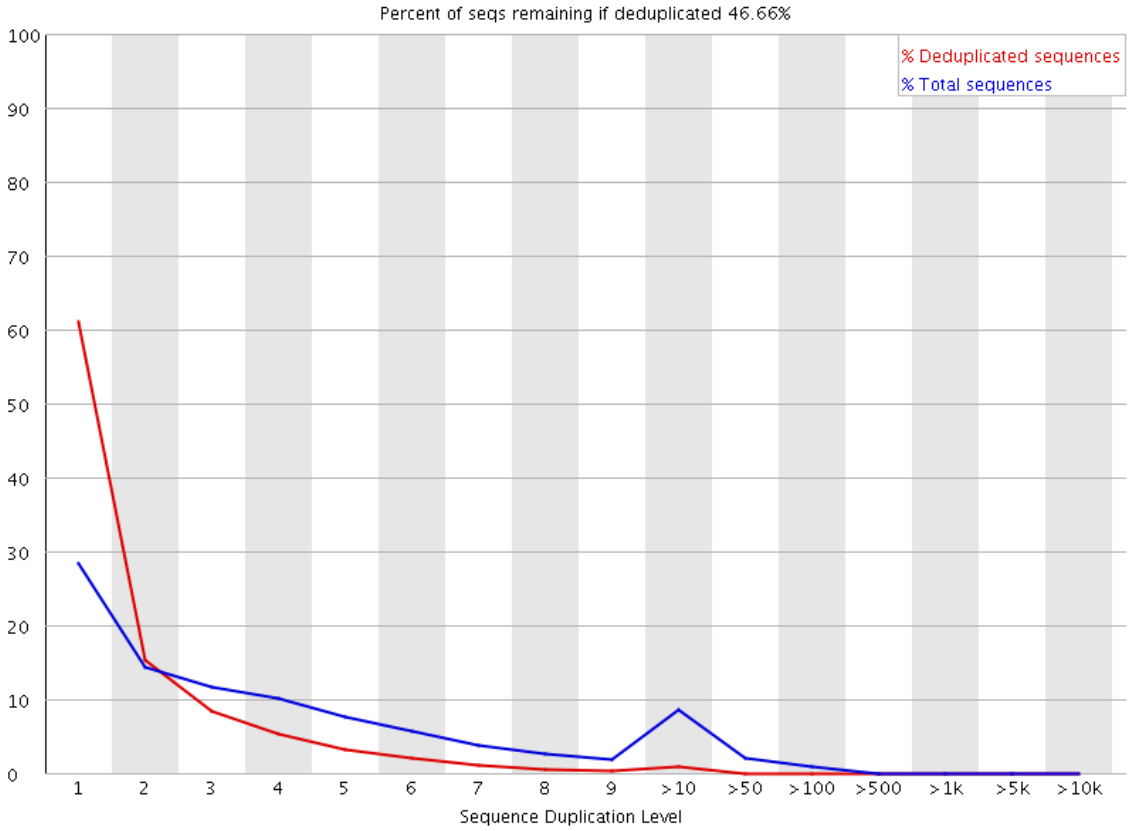


Figura 22 Duplicate sequences (Ejemplo)

Un bajo nivel de duplicación puede indicar un nivel muy alto de cobertura de la secuencia objetivo, pero un alto nivel de duplicación es más probable que indique algún tipo de sesgo de enriquecimiento.

Este módulo emitirá un error si las secuencias no únicas representan más del 50% del total y las secuencias no únicas representan más del 20% del total.

Overrepresented Sequences:

Este módulo enumera todas las secuencias que representan más del 0.1% del total. Para conservar la memoria, solo las secuencias que aparecen en las primeras 100,000 secuencias se rastrean hasta el final del archivo. Por lo tanto, es posible que este módulo pueda pasar por alto una secuencia que está sobrerrepresentada pero que no aparece al comienzo del archivo por alguna razón.

Este módulo emitirá un error si se encuentra que alguna secuencia representa más del 1% del total o una advertencia si se encuentra que alguna secuencia representa más del 0.1% del total.

Adapter Content:

Este módulo realiza una búsqueda específica de un conjunto de adaptadores definidos y da una visión de la proporción total de la librería que contiene estos adaptadores.

Este módulo emitirá una advertencia si alguna secuencia está presente en más del 10% de todas las lecturas o una advertencia si alguna secuencia está presente en más del 5% de todas las lecturas.

Kmer Content:

Este módulo mide el número de cada k-mero en cada posición en la librería y luego utiliza una prueba binomial para buscar desviaciones significativas de una cobertura uniforme en todas las posiciones. Se informa cualquier k-mero con enriquecimiento posicionalmente sesgado. Los 6 K-meros más sesgados se trazan grágicamente para mostrar su distribución.

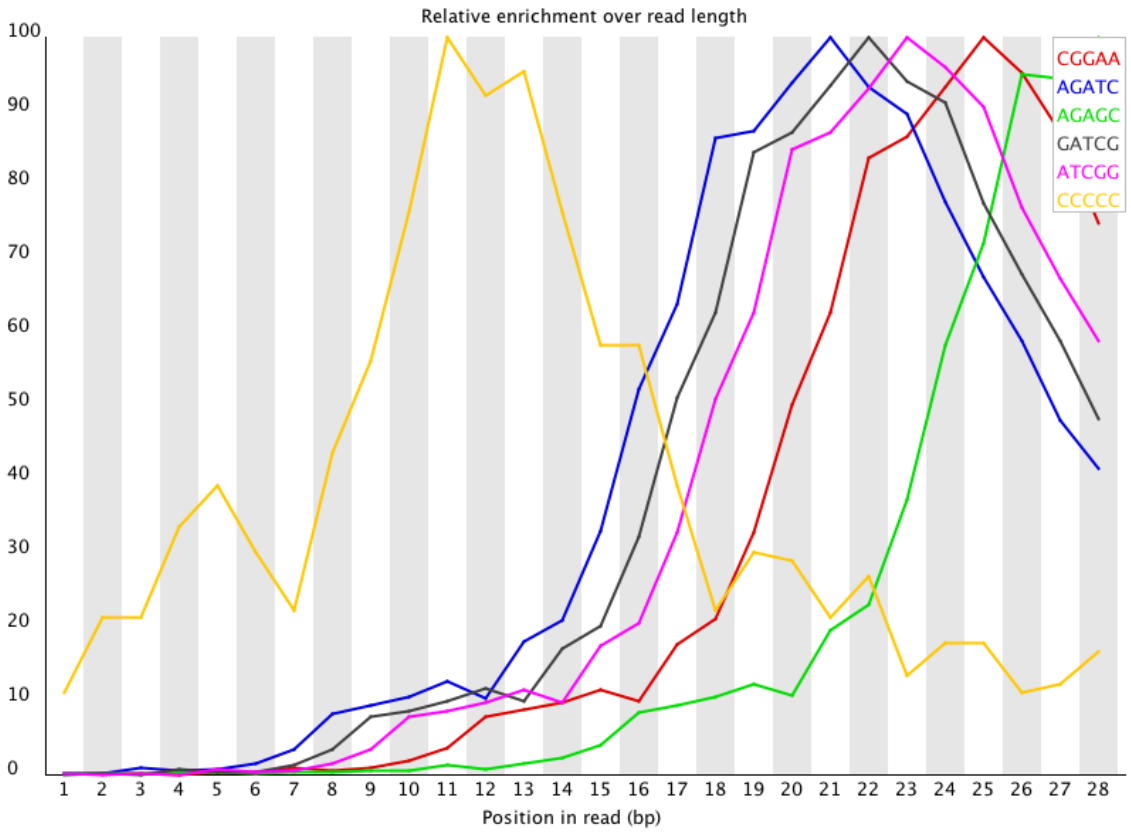


Figura 23 Kmer Content (Ejemplo)

Este módulo emitirá una advertencia si algún k-mero está desequilibrado con un valor p binomial $<10^{-5}$ o una advertencia si algún k-mero está desequilibrado con un valor p binomial <0.01.

Per Tile Sequence Quality:

Este gráfico solo aparecerá en los resultados del análisis si se utiliza una librería Illumina que conserva sus identificadores de secuencia originales. El gráfico muestra la desviación de la

calidad promedio de cada celda. Los colores están en una escala de frío a caliente, con colores fríos en posiciones donde la calidad es igual o superior al promedio de esa base en la ejecución, y los colores más cálidos indican que una celda tiene peores cualidades que otras para esa base.

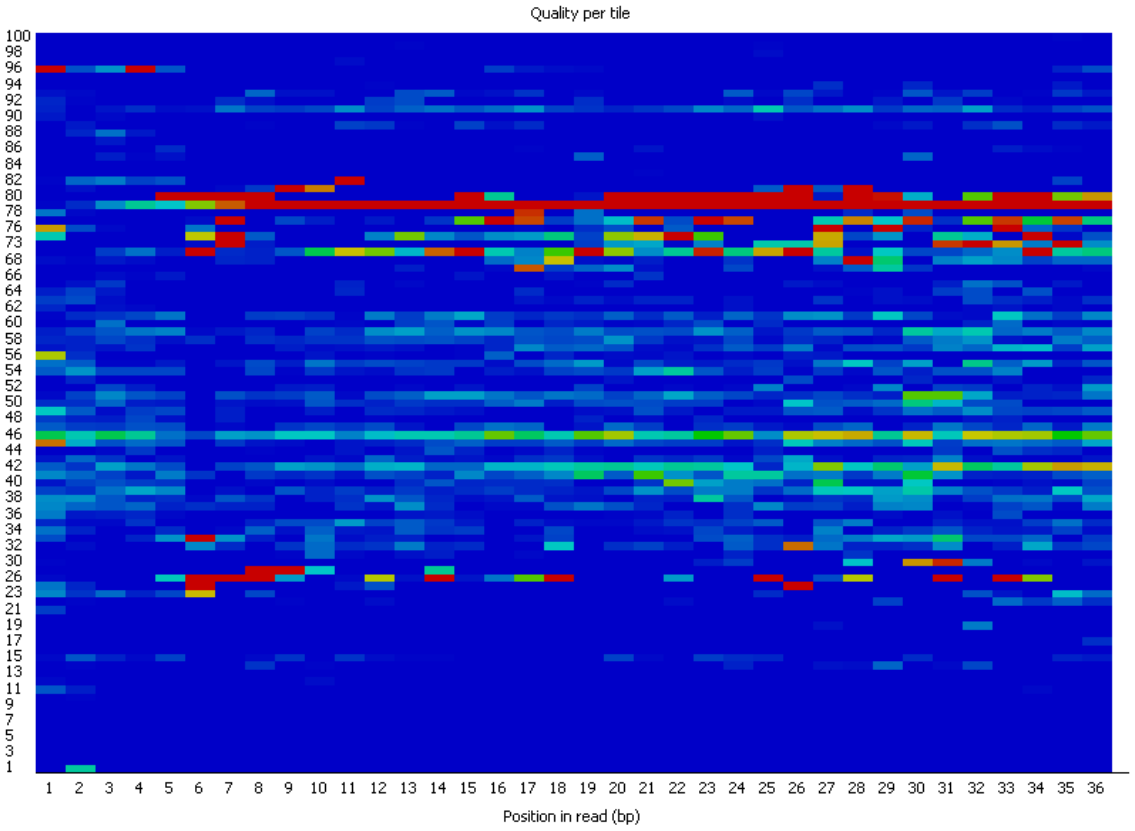


Figura 24 Per Tile Sequence Quality (Ejemplo)

Este módulo emitirá un error si alguna casilla muestra una puntuación media de Phred menos de 5 puntos que la media para esa base en todas las casillas o una advertencia si alguna casilla muestra una puntuación media de Phred menos de 2 puntos que la media para esa base en todas las casillas.

Opciones de configuración de Trimmomatic:

ILLUMINACLIP: corta adaptadores y otras secuencias específicas de Illumina de la lectura. Estas secuencias deben ser especificadas en un archivo con formato fasta. Y debe indicarse las condiciones de detección de las mismas en el siguiente orden :<fastaWithAdaptersEtc>:<seed mismatches>:<palindrome clip threshold>:<simple clip threshold>:<minAdapterLength>:<keepBothReads>

Donde <fastaWithAdaptersEtc> es la ruta al archivo que contiene los adaptadores, <seed mismatches> especifica el recuento máximo de discrepancias entre la lectura y el adaptador que permite realizar una coincidencia completa, <palindrome clip threshold> especifica cuán precisa debe ser la coincidencia entre las dos lecturas 'adaptadas ligadas' para la alineación de lectura del palíndromo PE, <simple clip threshold> especifica cuán precisa debe ser la coincidencia entre cualquier secuencia de adaptador, etc., contra una lectura, :<minAdapterLength> verifica que se haya detectado una longitud mínima de adaptador, y <keepBothReads> indicará que se conserven las lecturas pareadas.

- **SLIDINGWINDOW**: realiza un enfoque de recorte de ventana deslizante. Comienza a escanear en el extremo 5 'y recorta la lectura una vez que la calidad promedio dentro de la ventana cae por debajo de un umbral.
- **MAXINFO**: equilibra la longitud de lectura y la tasa de error para maximizar el valor de cada lectura
- **LEADING**: corta las bases al inicio de una lectura, si están por debajo de un umbral de calidad
- **TRAILING**: corta las bases al final de una lectura, si están debajo de calidad de umbral
- **CROP**: corta la lectura a una longitud especificada quitando bases del final de una lectura
- **HEADCROP**: corta el número especificado de bases desde el inicio de la lectura
- **MINLEN**: descarta la lectura si está por debajo de una longitud especificada
- **AVGQUAL**: descarta la lectura si la calidad promedio está por debajo del nivel especificado
- **TOPHRED33**: Convierte puntajes de calidad a Phred-33
- **TOPHRED64**: Convierte puntajes de calidad a Phred-64