Herramienta web para la priorización de variantes de tipo INDEL y SNV obtenidas mediante NGS.

Elisabet Castro Blanco

Máster en Bioinformática y Bioestadística

Bioinformática clínica

Joan Maynou Fernández

Javier Luís Cánovas Izquierdo

Junio del 2019

FICHA DEL TRABAJO FINAL

Título del trabajo:	Herramienta web para la priorización de variantes de tipo INDEL y SNV obtenidas mediante NGS
Nombre del autor:	Elisabet Castro Blanco
Nombre del consultor/a:	Joan Maynou Fernández
Nombre del PRA:	Javier Luís Cánovas Izquierdo
Fecha de entrega (mm/aaaa):	06/2019
Titulación::	Máster en Bioinformática y Bioestadística
Área del Trabajo Final:	Bioinformática clínica
Idioma del trabajo:	Español
Palabras clave	Variantes, Priorización, NGS

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

La facilidad para secuenciar ácidos nucleicos con un coste asequible y en un corto periodo de tiempo, ha facilitado la identificación de pequeñas variantes dentro del genoma. La gestión de todos los datos producidos para su aplicación requiere de conocimientos bioinformáticos. La falta de dichos conocimientos entre el personal sanitario, hace que las aplicaciones con interfaz gráfica que realizan la lectura, procesamiento y priorización cobren cada vez mayor importancia.

El objetivo de este trabajo es diseñar una aplicación web que permita realizar la priorización de variantes mediante una interfaz gráfica y sea capaz de trabajar con casos índice y trío. Para ello, se utiliza el lenguaje R para el procesamiento y priorización de las variantes, y el paquete Shiny para realizar la aplicación web interactiva.

Como resultado se obtiene una aplicación organizada en tres pestañas, que permite al usuario cargar sus archivos anotados, si se carga un caso trío se permite seleccionar un modelo de herencia. Después del procesamiento, mediante los parámetros de priorización implementados, es posible reducir en gran medida el número de variantes posible.

Como trabajo futuro a fin de mejorar la aplicación diseñada, se plantea aumentar el número de parámetros a controlar por el usuario y la posibilidad de exportar los resultados de la priorización en formato VCF.

Abstract (in English, 250 words or less):

The ease of sequencing DNA in a short period of time with an affordable cost, has facilitated the identification of small variants within the genome. The management of all this data requires bioinformatic knowledge. The lack of such knowledge among hospital staff, makes applications with graphical interface that performs prioritization very important.

The objective of this work is to create a web application that allows the prioritization of variants through a graphical interface. To do this, we had use the R language for the processing and prioritization of the variants, also we used the Shiny package to make the interactive web application.

The web application is organized in three tabs, which allows the user to upload their annotated files. If it's a familiar case, allows the user to select an inheritance model. After processing, through the prioritization parameters implemented, it is possible to greatly reduce the number of possible variants.

The future lines of work, in order to improve the application, are oriented to increase the number of parameters, that the user can control and the possibility of exporting the results of the prioritization in VCF format.

Índice

Lista de figuras

1. Introducción

1.1 Contexto y justificación del Trabajo

Gracias a las últimas tecnologías de secuenciación de ácidos nucleicos, actualmente es sencillo, rápido y medianamente asequible identificar diferentes variantes en el genoma humano.

Actualmente, existen diversos servidores públicos que contienen datos de secuenciación del genoma humano, datos sobre las diferentes variantes identificadas y multitud de anotaciones tanto de variantes como de genes humanos. Gracias a todos estos datos, es posible asociar variantes en el genoma con enfermedades, aunque no es nada fácil y a día de hoy sigue siendo un reto para la bioinformática.

El procesamiento de archivos con datos de variantes genómicas es complicado, debido a las grandes cantidades de información que contienen sobre variantes y sus anotaciones. Por este motivo, es necesario desarrollar algoritmos que sean capaces de filtrar estas variantes según sus anotaciones y de esta forma, sea posible deducir relaciones entre la aparición de determinadas variantes y la aparición de enfermedades.

La correcta priorización de variantes es vital para poder identificar candidatos a ser causantes de cierta enfermedad. Por tanto, conociendo que una enfermedad está relacionada con una o varias variantes, se podría realizar un diagnóstico precoz de la patología.

El manejo de estos datos en el ámbito clínico requiere de una aplicación con una interfaz gráfica, ya que en general, el personal hospitalario no tiene nociones de programación y tratamiento de datos. Por ello, el desarrollo de una aplicación web que realice la priorización de las variantes obtenidas es muy interesante para la aplicación a la clínica.

En este contexto, en este trabajo final de máster se pretende desarrollar una aplicación web que facilite la priorización y filtrado de estas variantes genéticas. Esta aplicación web realizará el filtrado de variantes génicas según una serie de parámetros de priorización, facilitando el diagnóstico y evitando los inconvenientes del proceso a nivel de procesamiento de datos. Teniendo la capacidad de trabajar tanto con casos índice (un único paciente) como con casos trío o familiar, en los que se tendrá en cuenta el modelo de herencia de la variante.

1.2 Objetivos del Trabajo

1.2.1. Objetivo principal

- Desarrollar una aplicación web para la priorización de variantes tipo INDEL y SNV obtenidas mediante NGS.

1.2.1. Objetivos secundarios

- Familiarizarse con los archivos VCF y sus anotaciones.
- Definir que parámetros se usarán para realizar la priorización de variantes.
- Diseñar la aplicación web.

1.3 Enfoque y método seguido

Los datos necesarios para este proyecto se han obtenido del servidor público "Genome in a bottle" [1], de donde se han descargado los archivos VCF tanto de casos índice como de casos trío o familiar.

El proyecto Genome in a bottle es un consorcio académico organizado por NIST, cuya prioridad es caracterizar de genomas humanos para su uso como validación analitica y el desarrollo, optimización y prueba de diferentes tecnologías de secuenciación. Los datos obtenidos provienen de estudios de tipo Whole Genome, que se ha realizado usando 5 plataformas de secuenciación diferentes: Illumina, Complete Genomics LFR, 10X Genomics GemCode WGS, Ion Proton exome y SOLiD [2].

Los archivos obtenidos están sin anotar, por lo que se han descargado 4 bases de datos diferentes para realizar las anotaciones: gnomAD [3], dbSNP [4], dbNSFP [5] y ClinVar [6]. El proceso de anotación es indicar la información asociada a la variante, para ello usaremos la aplicación snpEFF [7].

Para realizar la priorización, se ha realizado una selección de parámetros presentes en las anotaciones realizadas con las diferentes bases de datos.

Se diseña la interfaz gráfica del aplicativo web ,en la que se incluye tres pestañas dedicadas respectivamente a la subida de archivos por parte del usuario, la selección de parámetros para priorización en forma de desplegables, campos numéricos y de texto, y finalmente, la presentación de los resultados por medio de tablas. Para el diseño de la aplicación e implementación de todos los procesos necesarios se usa el lenguaje R junto con el paquete de R Shiny [8].

1.4.1. Tareas

Las tareas se han agrupado según las PECs marcadas en la planificación del TFM del aula.

- PEC 0: Definición de los contenidos del trabajo
    - Búsqueda de información
    - Lectura del material
    - Redacción y confección de los contenidos a incluir
- PEC 1: Plan de trabajo
    - Diseño del proyecto
    - Planificación de las tareas
- PEC 2: Desarrollo del trabajo. Fase 1
    - Obtención y organización de datos
    - Definición de los parámetros para la priorización
    - Diseño del algoritmo
- PEC 3: Desarrollo del trabajo. Fase 2
    - Diseño de la aplicación
    - Implementación del algoritmo
    - Comprobación del funcionamiento
- PEC 4: Cierre de la memoria
- PEC 5a: Elaboración de la presentación
- PEC5b: Defensa pública

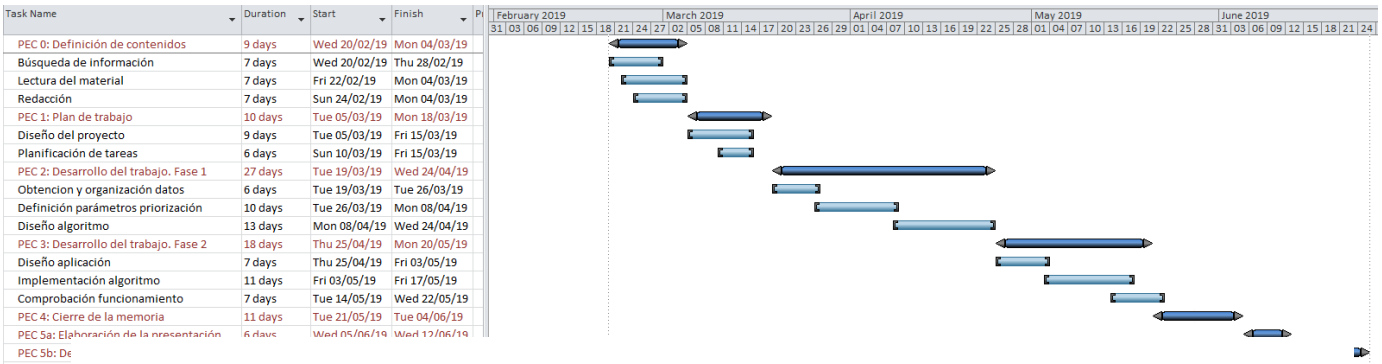
1.4.2. Hitos

- Inicio de la PEC - 0 20/02/2019
    - Inicio de la etapa de documentación sobre la temática. 21/02/2019
    - Definición del tema y contenido del trabajo
- Fin de la PEC 0 - 04/03/2019
- Inicio de la PEC 1 - 05/03/2019
    - Fin de la etapa de documentación sobre la temática. 05/03/2019
    - Planificación del proyecto y sus tareas
- Fin de la PEC 1 – 18/03/2019
- Inicio de la PEC 2 – 19/03/2019
    - Fin de la obtención y organización de los datos necesarios para el trabajo – 24/03/2019
    - Fin de la comprobación de las anotaciones – 26/03/2019
    - Fin de la definición de los parámetros que se usarán para realizar la priorización de variantes – 08/04/2019
    - Fin del diseño del algoritmo para la priorización – 24/04/2019
- Fin de la PEC 2 – 24/04/2019
- Inicio de la PEC 3 – 25/04/2019
    - Fin del diseño de la aplicación – 03/05/2019

    - Fin de la implementación del algoritmo en la aplicación – 03/05/2019
    - Fin de las comprobaciones de funcionamiento – 22/05/2019
- Fin de la PEC 3 – 20/05/2019
- Inicio de la PEC 4 – 21/05/2019
    - Cierre de la memoria
- Fin de la PEC 4 – 04/06/2019
- Inicio de la PEC 5a – 05/06/2019
    - Preparación de la presentación
- Fin de la PEC 5 – 12/06/2019
- Inicio de la PEC 6 – 13/06/2019
    - Defensa pública del TFM
- Fin de la PEC 6 – 25/06/2019

1.4.3. Diagrama de Gantt

El diagrama de Gantt correspondiente a la planificación del proyecto se muestra en la *Figura 1*.



Task Name	Duration	Start	Finish
PEC 0: Definición de contenidos	9 days	Wed 20/02/19	Mon 04/03/19
Búsqueda de información	7 days	Wed 20/02/19	Thu 28/02/19
Lectura del material	7 days	Fri 22/02/19	Mon 04/03/19
Redacción	7 days	Sun 24/02/19	Mon 04/03/19
PEC 1: Plan de trabajo	10 days	Tue 05/03/19	Mon 18/03/19
Diseño del proyecto	9 days	Tue 05/03/19	Fri 15/03/19
Planificación de tareas	6 days	Sun 10/03/19	Fri 15/03/19
PEC 2: Desarrollo del trabajo. Fase 1	27 days	Tue 19/03/19	Wed 24/04/19
Obtencion y organización datos	6 days	Tue 19/03/19	Tue 26/03/19
Definición parámetros priorización	10 days	Tue 26/03/19	Mon 08/04/19
Diseño algoritmo	13 days	Mon 08/04/19	Wed 24/04/19
PEC 3: Desarrollo del trabajo. Fase 2	18 days	Thu 25/04/19	Mon 20/05/19
Diseño aplicación	7 days	Thu 25/04/19	Fri 03/05/19
Implementación algoritmo	11 days	Fri 03/05/19	Fri 17/05/19
Comprobación funcionamiento	7 days	Tue 14/05/19	Wed 22/05/19
PEC 4: Cierre de la memoria	11 days	Tue 21/05/19	Tue 04/06/19
PEC 5a: Elaboración de la presentación	6 days	Wed 05/06/19	Wed 12/06/19
PEC 5b: De			

Figura 1: Diagrama de Gantt

1.5 Breve sumario de productos obtenidos

- Aplicativo web para la priorización de variantes: consta de una archivo (app.R), que contiene todas las partes (ui y server) necesarias para el funcionamiento de la aplicación web.
- Repositorio en Github: donde se encuentra la aplicación disponible para su descarga y la información acerca de la misma.
- Memoria Final del trabajo: que recoge y amplía los documentos entregados en las tres primeras PECs

1.6 Breve descripción de los otros capítulos de la memoria

❖ **Estado del arte**: se resumen las técnicas de secuenciación desarrolladas a lo largo de la historia, las cuales han permitido que se identifiquen las

variantes génicas que son el objeto de este trabajo. También se incluye un resumen de las diferentes aplicaciones ya existentes para realizar la priorización de variantes.

- **Datos de partida**: se explica de dónde se obtienen los datos con los que se ha realizado el trabajo, indicándose su formato y el procesamiento necesario para ser incluidos en este trabajo.
- **Anotación de los ficheros**: se expone como se realizó el proceso de anotación de los diferentes archivos y qué bases de datos se utilizaron
- **Selección de los parámetros de priorización**: se enumeran y explican los parámetros de priorización escogidos para la aplicación web y se resume cómo serán implementados en la aplicación.
- **Diseño de la aplicación web**: se expone como se ha diseñado la aplicación web, incluyendo las herramientas usadas y las diferentes opciones que se contemplaron durante la realización del proyecto.
- **Procesamiento de los VCFs subidos por el usuario**: se explica de qué manera la aplicación web procesa los archivos, indicando las diferentes fases y como se llega a la tabla que será el punto de partida para la priorización.
- **Priorización**: se define en detalle cómo se ha implementado la priorización de las variantes y las funciones que realizan los diferentes botones presentes en este apartado de la aplicación web.
- **Aplicación final**: se explica detalladamente la estructura de la interfaz gráfica y las funciones disponibles en la aplicación.
- **Conclusiones**: se exponen todas las conclusiones extraídas durante la realización del TFM.
- **Glosario**: se definen términos usados durante la memoria del TFM
- **Bibliografía**: se incluyen todas las fuentes que se han consultado para la realización del TFM.
- **Anexos**

2. Resto de capítulos

2.1. Estado del arte

2.1.1. Tecnologías de secuenciación y aplicaciones

Los métodos de secuenciación han supuesto un cambio sustancial en la forma de entender la organización de la información biológica dentro de los organismos, abriendo un nuevo campo para la investigación.

El primer método de secuenciación desarrollado fue el método de Sanger [9], en el que se usaban nucleótidos normales y dideoxi-nucleótidos, de forma que cuando la ADN polimerasa incorporaba uno de estos últimos, al no tener el extremo 3' con un grupo hidroxilo libre, la cadena terminaba. Se realizan 4 reacciones añadiendo en cada una de ellas un dideoxi-nucleótido. Después, el resultado de las cuatro reacciones (cada una en un pocillo) se cargaba en una electroforesis desnaturalizante con gel de poliacrilamida, se obtiene un patrón de bandas que permite averiguar la secuencia de la molécula de DNA *(Figura 2)*.

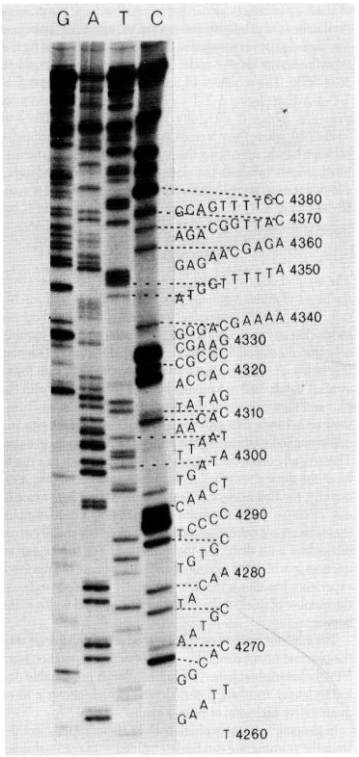


Figura 2: El método de Sanger.

Gel de poliacrilamida, en el que se puede deducir la secuencia de una molécula de ADN. Imagen extraída de: F. Sanger et al. (1977)

Las limitaciones de esta técnica de secuenciación son principalmente que está limitada a fragmentos de 200 pares de bases, su alto coste económico en reactivos, ya que es necesario para secuenciar un único fragmento realizar 4 reacciones, en cada una añadiendo un dideoxi-nucleótido, por lo que es proceso lento.

A partir de este método con algunas mejoras, como el marcado fluorescente de los dideoxi-nucleótidos que permitió realizar todas las reacciones en un mismo tubo [10], se desarrollaron secuenciadores automáticos con las siguientes características [11]:

- Es un método con una alta precisión.
- Está restringido a un fragmento de DNA con una longitud máxima de 1000 pares de bases .
- Tiene un alto coste económico.

Este método se usó para el proyecto Genoma Humano [12], este proyecto pretendía secuenciar la totalidad del genoma humano, en un tiempo aproximado de 15 años y un coste aproximado de tres mil millones de dólares.

A principios de la década de los 2000, comenzaron las técnicas de alto rendimiento que permitieron secuenciar muchísimos genomas, lo que permitió realizar estudios de asociación de genoma completo. En estos estudios, se empezaron asociar genes con enfermedades complejas y permitiendo la detección y screening de SNPs (Single Nucleotide Polimorphisms) [11].

A continuación, empezaron las NGS (Next Generation Sequencing), este término engloba a todas las técnicas que permiten un secuenciado masivo de ácidos nucleicos, como por ejemplo la pirosecuenciación y el secuenciado por hibridación. Las principales ventajas de estas técnicas de nueva generación son el menor coste económico del proceso y su rapidez. Al suplir defectos de las tecnologías anteriores, la detección de variantes se facilitó. Se introdujo la secuenciación guiada, que consiste en poder seleccionar la parte del genoma a secuenciar, reduciendo aún más los costes [11].

La pirosecuenciación [13] consta de los siguientes pasos: primero se fragmenta el ácido nucleico a secuenciar, a continuación a los fragmentos se les ligan los adaptadores que son pequeñas oligonucleótidos de secuencia conocida. Se realiza la amplificación clonal y la secuenciación se produce en micropocillos , en los que se van ofreciendo los diferentes nucleótidos, de forma que cuando el nucleótido aportado es incorporado a la molécula de ADN naciente, se libera un pirofosfato ($P_2O_7^{4-}$ o PPi). Este pirofosfato es convertido en luz por medio de dos reacciones, la luz emitida es captada por una cámara CCD que procesa los datos de todos los pocillos en cada ciclo.

En la secuenciación por hibridación [14] se usan nucleótidos terminadores marcados con moléculas fluorescentes igual que en el método mejorado de Sanger, pero a diferencia de Sanger permite realizar secuenciación paralela masiva, es decir, se pueden realizar millones de secuencias a la vez. Otra diferencia importante respecto a los métodos anteriores es la posibilidad de quitar el fluoróforo (molécula fluorescente) después de haber captado la fluorescencia, liberando de esta forma el extremo 3’ de la molécula, lo que hace que sea un terminador reversible. Como novedad, en esta técnica, la amplificación y secuenciación se realiza sobre una superficie sólida.

Actualmente, las NGS se están utilizando para comprobar predisposiciones a ciertas enfermedades, para predecir el funcionamiento de fármacos, descubrir nuevas dianas terapéuticas, diagnóstico de enfermedades monogénicas entre otras muchas cuestiones.

Gracias a las técnicas de secuenciación, se describieron variantes de pequeño tamaño, son aquellas que implican la variación de un número reducido de nucleótidos, como por ejemplo SNVs, INDELs que podrían estar asociadas a enfermedades. Estas variantes se pueden detectar de diferentes formas: paneles custom [15], WES (Whole Exome Sequencing) que son estudios en los que se secuencia el exoma, es decir, secuenciar todos los genes que codifican

para proteínas, muy importantes para identificar genes relacionadas con enfermedades monogenéticas y WGS (Whole Genome Sequencing) [11].

En este trabajo nos centraremos en las variantes de tipo SNV (Single Nucleotide Variant) y tipo INDEL (pequeñas inserciones y deleciones). El conocimiento de estas variantes y sus efectos clínicos podría ser de utilidad de cara a la realización de screenings en los cuales buscamos diagnosticar ciertas enfermedades, a la emergente medicina personalizada, entre otras muchas aplicaciones [15]. Por tanto, la detección y anotación de variantes junto con la priorización es de vital importancia para poder extraer información útil de los datos obtenidos con las tecnologías de secuenciación actuales y poder seguir avanzando en la investigación de la etiología de multitud de enfermedades [16].

Durante un estudio de secuenciación de genoma completo, se pueden llegar a encontrar varios cientos de miles de variantes genómicas. Además, hay que tener en cuenta la complejidad genética de las enfermedades, ya que rara vez están causadas por un único gen (enfermedades monogenéticas). Una buena priorización es clave para llegar a un diagnóstico correcto, ya que permite filtrar las variantes encontradas durante la secuenciación, y extraer aquellas que realmente pueden indicarnos que la existencia de alguna patología.

2.1.2. Aplicaciones existentes para la priorización de variantes

En la actualidad existen una gran variedad de aplicaciones y algoritmos publicados por diversos grupos de investigación que intentan resolver o abordar este problema. Una parte importante de las aplicaciones desarrolladas se centran en la detección e identificación de variantes, tanto INDEL como SNV, como por ejemplo: VarDict [17] y SNVSniffer [18]. SiNVICT que es capaz de detectar e identifica variantes SNV. Además, es capaz de a partir de diferentes muestras de un mismo individuo a diferentes tiempos, realizar análisis de líneas temporales y saber si se han producido mutaciones en el tiempo de estudio. Esta utilidad está especialmente pensada para analizar muestras de ADN tumoral circulante en casos de pacientes oncológicos [19]. También, hay aplicaciones especializadas en filtrar variantes, como GEMINI [20], Varapp [21] y BrowseVCF [16], en agrupar variantes originadas por inserciones y deleciones, como VarGrouper [22]. El gold standard es GATK, centrado en la identificación de variantes SNPs e INDELs en la línea germinal. Esta herramienta ha sido desarrollada por la Plataforma de Data Sciences en Broad Institute. Ofrece gran variedad de herramientas para la identificación y genotipado de variantes [23].

Respecto a las aplicaciones existentes para filtrar variantes, Varapp permite el filtrado por características familiares, distinguiendo gráficamente los pacientes que tienen que variante y si el individuo es portador de dicha variante. En cuanto a velocidad de procesamiento esta aplicación es más rápida que GEMINI. El punto débil de esta aplicación es que no permite importar datos desde archivos VCF. En cambio, la aplicación BrowserVCF si tiene esta funcionalidad, permitiendo seleccionar que variantes del archivo dado queremos importar. Este

detalle, permite que el algoritmo para filtrar devuelva los resultados de la consulta más rápido.

La aportación de este TFM es una aplicación con una interfaz gráfica sencilla, capaz de realizar la lectura y procesamiento de VCFs anotados, extrayendo información relevante y mostrandola al usuario de forma ordenada. La priorización se ha implementado de forma que el usuario no tiene por que conocer como se expresa esa característica en el archivo VCF. El aplicativo realiza la priorización en unos pocos segundos, siendo el proceso más lento la lectura y el procesamiento inicial.

La falta de personal formado en bioinformática en los centros hospitalarios y en el ámbito clínico en general, supone un problema a la hora de aplicar el análisis genético y de variantes en la clínica. Debido a este problema, las aplicaciones capaces de manejar este tipo de datos, realizando la priorización a través de una interfaz gráfica, tienen cada vez mayor importancia.

2.2. Datos de partida

Los archivos VCFs de partida son descargados del servidor público “Genome in a bottle” [1], es un consorcio académico organizado por NIST, dedicado al desarrollo de la infraestructura técnica que permita la traducción de la secuenciación de genoma completo a la clínica. Como se ha comentado en el capítulo *1.3 Enfoque y método seguido*, los datos obtenidos provienen de estudios de tipo Whole Genome, que se han realizado usando 5 plataformas de secuenciación diferentes [2].

Los datos que se han descargado corresponden con la última versión de los archivos (en el momento de la realización de este trabajo). La versión del archivo utilizado para el caso índice es NA12878_HG001. En el caso trío, se ha usado la versión HG002_NA24385 del archivo correspondiente con el hijo, la versión HG003_NA24149 del archivo correspondiente con el padre, y por último la versión del archivo HG004_NA24143 correspondiente con la madre, todos de la última versión del trio Ashkenazim. Todos los archivos están disponibles en el siguiente enlace.

Los archivos obtenidos son VCFs que contienen las diferentes variantes encontradas al secuenciar el genoma completo, la versión del ensamblado del genoma humano usada en estos archivos es GRCh37 (hg19). El motivo para escoger los archivos en esta versión del genoma, a pesar de que estén disponibles en versión hg38, es que la mayoría de las bases de datos están bien desarrolladas en hg19.

2.3. Anotación de los ficheros

Puesto que los datos de “Genome in a bottle” no estaban anotados, se procedió a bajar cuatro bases de datos diferentes para realizar la anotación usando snpEFF [7]. El comando usado para la anotación con dicho programa es:

```
java -Xmx8g -jar Ruta\snpEff\SnpSift.jar annotate -a Ruta\BaseDatos.vcf Ruta\Archivo_NO_Anotado.vcf > Ruta\Archivo_Anotado.vcf
```

Las bases de datos escogidas fueron: gnomAD (versión: r2.1.1) [3], dbSNP (versión: build 152) [4], dbNSFP (versión: v4.0b2) [5] y ClinVar (versión: 2019-03-25) [6].

La base de datos gnomAD aporta recuentos y probabilidades de la variante en distintos conjuntos de personas, diferentes razas según el lugar donde viven. La base de datos dbSNP está especializada en variaciones pequeñas como SNV (Single Nucleotide Variant), pequeñas INDEL o pequeñas repeticiones en támdem, recoge e identifica asignando códigos alfanumericos (“rs” seguido de un número) a las diferentes variantes, también permite acceder a otras bases de datos mediante links. La base de datos sbNSFP aporta las puntaciones de diferentes algoritmos de predicción de características como la patogeneicidad de las variantes. La base de datos ClinVar aporta datos sobre la implicación de las variantes en procesos patológicos y la relación de las variantes con el fenotipo.

Para realizar la anotación debido al gran tamaño del archivo, se realizó con snpEFF una partición del archivo, dividiendolo en tantas partes como cromosomas *(Figura 3).*

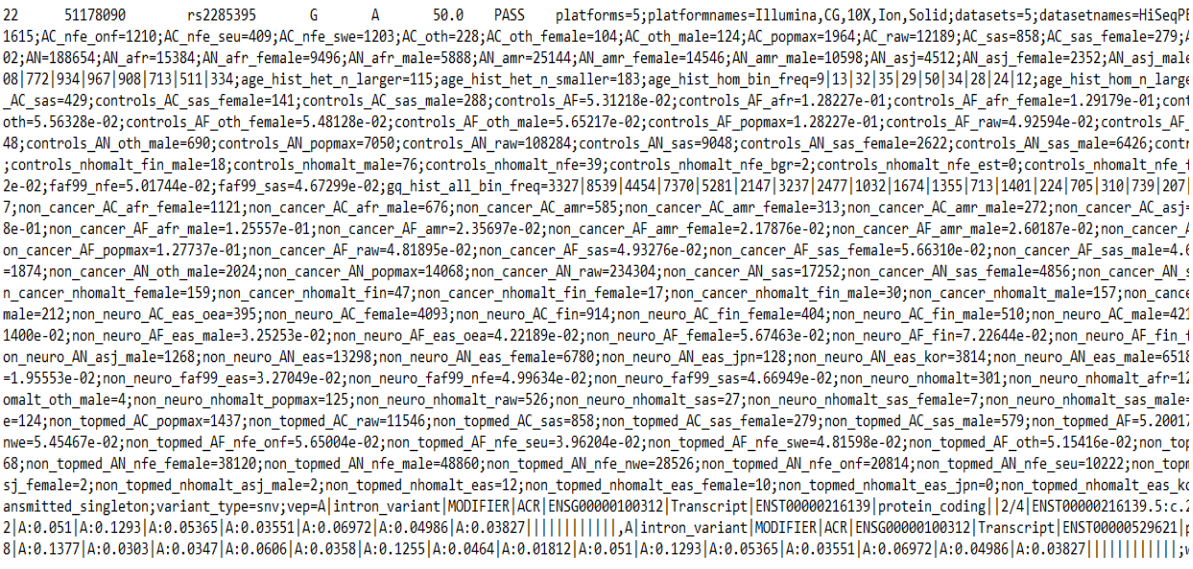


Figura 3: Vista de una variante después del proceso de anotación

En esta figura podemos ver el aspecto de una variante anotada en el archivo VCF del cromosoma 22 del caso índice.

2.4. Selección de los parámetros de priorización

Para seleccionar que campos de la columna Info del VCF se iban a usar como parámetros para priorizar las variantes se ha pensado desde un punto de vista clínico, en el que el usuario puede ser un médico o un genetista, por lo que, la priorización se hace mediante desplegables, cuadros numéricos y cuadros de texto. Esta aplicación está centrada en las variantes de la línea germinal.

Los parámetros de priorización escogidos son los siguientes:

- Tipo de variante: indica de que tipo es la variante. Ejemplos: Single Nucleotide Variant (snv), insertion-delection (INDEL), etc.
- Qual: calidad de la secuenciación en phred scale[1].
- DP: es la profundidad de lectura (Deep read).
- Genotipo: con (|) o sin información de fase (/), donde 0 es la referencia, 1 es el primer alelo alternativo en ALT.
- Nombre del gen: indica si procede el nombre del gen al que afecta la variante.
- Proteína (nombre de entrada Uniprot): indica a qué proteína afecta.
- Enfermedad asociada: en este campo se especifica en qué enfermedad está implicada la variante.
- Efecto: se refiere a si la variante es benigna o maligna.
- Frecuencia europea: indica la frecuencia del alelo mutado en muestras de ascendencia europea.
- Impacto en la proteína: indica que la magnitud del impacto que se produce en la proteína resultante. Los grados del impacto en la proteína son: alto, moderado, bajo y modificadora.
    - Alto: cuando la variante tiene consecuencias drásticas para la proteína final. Ejemplo: la variante añade un codón de stop prematuro. Cuando el mRNA se traduce, el codón de stop añadido produce una interrupción de la traducción y la proteína queda incompleta. Estas proteínas truncadas suelen perder la función que tenía la proteína original.

[1] **Phred scale:** es una medida de la calidad de la identificación de nucleótidos por sistemas de secuenciación automática, está logarítmicamente relacionada con la probabilidad de que una base sea identificada de forma errónea. Actualmente, es muy usada para saber la calidad de las secuencias de ADN y es la base para la comparación de diversos métodos de secuenciación. Por ejemplo: Pred scale = 30 supone que la precisión es 99.9%.

- Moderado: la variante provoca cambios en la región codificante, que pueden ser deleción o inserción de unas pocas bases o cambios de una base por otra que produzcan mutaciones no sinónimas.
- Bajo: en este grado se incluyen las mutaciones sinónimas, cambios en el codón de comienzo y fin que mantienen estos codones o cambios en la región de splicing.
- Modificadora: la variante tiene un efecto modificador importante sobre la función de la región donde se encuentra.

2.5. Diseño de la aplicación

Desde el comienzo del proyecto, la idea sobre la aplicación web ha sido la siguiente:

Primero, el aplicativo web pide que subamos los archivos anotados a procesar, pudiendo ser estos de un caso índice o de un caso familiar/trio. En el segundo paso, la aplicación web permite seleccionar valores para una selección de 10 parámetros comunes a ambos casos posibles (índice o trio), de forma que el resultado final irá cambiando según el usuario vaya introduciendo valores en los distintos filtros. Por último, se muestran los resultados en una tabla después de la lectura y filtrado de los archivos aportados.

R es un lenguaje de programación para computación estadística y una gran variedad de gráficos. Actualmente es uno de los más usados en campos de investigación. Tiene mas de 2000 paquetes disponibles en su repositorio (CRAN) y es compatible con los principales sistemas operativos.

Rstudio es un entorno para R, que incluye una consola, un editor de sintaxis, el cual admite la ejecución de código, y herramientas para gestionar todo lo relacionado con el espacio de trabajo. Además, a través de dicha aplicación se pueden generar documentos Rmarkdown con salidas a PDF, documentos de texto o documentos HTML, entre otras muchas cosas.

Para el diseño y la implementación de esta aplicación web se ha usado el paquete “Shiny” de R [8]. Se utiliza el paquete “vcfR” [24], para el procesamiento de los VCFs crudos subidos a través de la interfaz. Y por último, para mostrar los resultados se usa el paquete “DT” [25], que permite acceder a la librería de tablas de JavaScript.

Los archivos VCF son archivos de texto usados para anotar y almacenar datos sobre variantes genéticas. Estos archivos poseen una estrutura bien definida formada por: una cabecera y el cuerpo del archivo. La cabecera del archivo contiene los metadatos que describen todos los campos del resto del archivo, cada línea de estos metadatos comienza por ##. En el cuerpo del archivo se

describen las diferentes variantes, una por cada línea. Estas filas están divididas en 8 columnas principales:

- CHROM: se indica el cromosoma donde se encuentra la variante.
- POS: indica la posición dentro del cromosoma.
- ID: indica el identificador de la variante.
- REF: contiene la base o bases que se encuentran en esa posición originalmente.
- ALT: indica por que base o bases se han cambiado las originales.
- QUAL: es un número que es una medida de la calidad asociada con la inferencia de los alelos.
- FILTER: indica si la variante ha pasado una serie de filtros.
- INFO: contiene una lista más o menos extensa de anotaciones que describen la variante.

El paquete vcfR permite el procesamiento de archivos VCF, que por las funciones de R base no sería posible leer, debido al gran encabezamiento típico de este tipo de archivos. Como resultado de la lectura inicial, se obtiene un objeto de tipo vcfR, que contiene entre otros dos matrices correspondientes con el campo GT (dos columnas, la primera indica el formato y la segunda es el contenido del campo GT), y los datos de las variantes del archivo (8 columnas, siendo la última el campo INFO, donde se encuentran todas las anotaciones). A partir de estas matrices, se crean los data frames con los datos de las variantes.

El paquete Shiny permite la creación de aplicaciones interactivas basadas en el lenguaje de R. Este paquete permite la programación reactiva [26], en la cual todos los objetos son susceptibles de cambiar en respuesta a las acciones del usuario, y a los inputs introducidos por el mismo. Esta propiedad, otorga una gran capacidad de control a la hora de establecer los parámetros para la priorización.

Todas las aplicaciones realizadas en Shiny constan de dos partes bien diferenciadas:

1. Ui.R: incluye todo el código referido a la interfaz gráfica de la aplicación diseñada.
2. Server.R: incluye el código encargado del procesamiento de los datos y realización de las diferentes tablas, figuras, etc.

En este caso, en lugar de usar dos archivos correspondientes a las partes citadas en el párrafo anterior, se ha optado por usar un único archivo que agrupa ambas partes. De esta forma, el usuario solo tiene que abrir este archivo (app.R) y hacer clic en “Run App”, facilitando en gran medida el lanzamiento de esta.

En cuanto a la interfaz gráfica del aplicativo web, en un principio se pensó en mostrar todo en una única pestaña. Esta idea fue descartada, ya que resultaba demasiado densa. Después, se realizó un planteamiento de la aplicación con 4

pestañas, donde una de ellas, contenía la explicación de los parámetros de priorización disponibles. La información de esta pestaña se decidió finalmente poner en la misma pestaña que los parámetros de priorización, dando lugar a las tres pestañas que tiene la aplicación final (información ampliada en 2.8. Aplicación final).

Todo el proceso se ha realizado en la aplicación Rstudio versión 1.1.463, basado en el software de programación open source R versión 3.5.1.

2.6. Procesamiento de los VCFs subidos por el usuario

2.6.1. Caso índice

Cuando el usuario sube un archivo VCF en la sección dedicada a casos índice, el procesamiento de este vcf se realiza con ayuda del paquete de R “vcfR”, que nos permite leer el archivo extrayendo la información de las diferentes variantes en forma de matrices de datos. La columna Info requiere un tratamiento especial, que permitirá separar los distintos campos presentes en esta columna. Al final del proceso de lectura, obtenemos tres tablas. La primera contiene las columnas principales del vcf (Chrom, Pos, ID, ref, alt, QUAL, filter e info), la segunda tabla obtenida contiene el campo GT. Y por último, en la tercera tabla, tenemos la columna info con todos sus campos separados.

Con el objetivo de que las columnas que aparecen en los resultados sean más cómodas de consultar (user-friendly). En muchos casos la información se encuentra agrupada en un único campo, por lo que estos campos han sido parseados para extraer la información que se desea mostrar, eliminando de esta manera información adicional que dificulta la lectura de la misma.

El proceso de lectura y procesamiento es el proceso más lento y que más recursos computacionales requiere de la aplicación. Por ello se ha incluido una barra de progreso en la esquina inferior izquierda, que permite ver el avance del proceso según se completan los diferentes pasos.

Toda la información se agrupa en una única tabla, que será la base para el siguiente paso, la priorización.

2.6.2. Caso trío

En el caso del trío, se sigue el mismo proceso explicado para el caso índice con cada uno de los archivos subidos. Por lo que, es un proceso bastante lento que depende en gran medida del tamaño de los archivos.

Además en este caso, se aplican diferentes modelos de herencia antes de pasar a la priorización. Los modelos de herencia contemplados en el aplicativo web son: variantes autosómicas dominantes heredadas de los parentales y originadas de novo, variantes autosómicas recesivas heredadas de los

parentales y de novo, variantes ligadas a X dominantes heredadas y de novo, y por último variantes ligadas a X recesivas heredadas y de novo. En el caso, de trabajar con variantes ligadas a X, se pide indicar el sexo del hijo/ hija. Con objeto de facilitar la selección del modelo de herencia, en la aplicación se han implementado en forma de cuestionario con desplegables *(**¡Error! No se encuentra el origen de la referencia.**)*, lo que permite el uso de la aplicación por usuarios que no sepan como se codifican los diferentes genotipos en los archivos VCF.

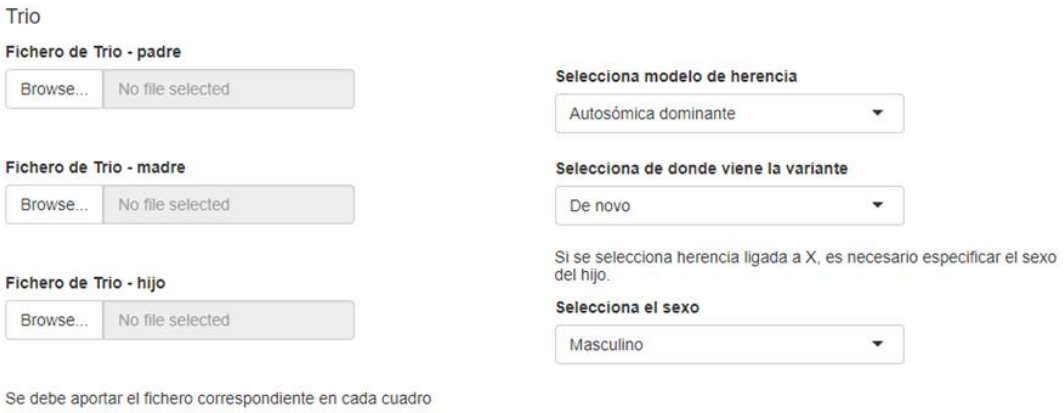


Figura 4: Subida de archivos. Sección Tríos

A continuación, veremos los diferentes modelos de herencia:

- Autosómica dominante de novo: una variante de novo es aquella que no está presente en los parentales, pero sí en el hijo. La aplicación busca variantes presentes en el hijo y no en los padre que tengan genotipo 0/1.

- Autosómica dominante heredada: uno de los parentales tiene la variante en heterocigosis y al ser dominante, este parental está afectado. En este caso, la aplicación buscará variantes que estén presentes en uno de los parentales y en el hijo, con genotipo 0/1 en los parentales y 0|1 en el hijo.

- Autosómica recesiva de novo: cuando se sigue este modelo, como es muy raro que muten de novo las dos copias, se considera que una de las variantes viene de alguno de los parentales, mientras que la otra variante ha sido originada de novo. La aplicación busca variantes en el hijo con genotipo 1/1, y que no esté en los parentales variantes con genotipo 0/1 que también estén presentes en el hijo.

- Autosómica recesiva heredada: en este modelo de herencia se considera que la variante está presente en ambos parentales, y que ambos se la han transmitido al hijo, por lo que este es homocigótico para esta variante. En este caso, la aplicación buscara variantes con genotipo 0/1 en los

parentales compartidas con el hijo, y además extraerá las variantes del hijo con genotipo 1/1.

- Ligada a X dominante de novo:
    - Hija: en este caso, se buscarían variantes que solo estén presentes en la hija con genotipo 0/1, pero que no estén en los parentales.
    - Hijo: el hijo tendría solo la copia del gen con la variante, por lo que el genotipo sería 1/1. Esta variante no tendría que estar presente en ninguno de los parentales.
- Ligada a X dominante heredada:
    - Hija: en este caso, se buscan variantes dentro del archivo de la hija que tengan genotipo 0/1. Si está variante se ha heredado, tendrá que estar presente en alguno de los parentales, por lo que buscaremos variantes con genotipo 0/1 en la madre y 1/1 en caso del padre, que compartan los parentales y la hija.
    - Hijo: para que el hijo tenga la variante, la madre debe tenerla también (cromosoma X de los hijos viene de la madre). Por ello, se buscarán variantes con genotipo 1/1 en el hijo y variantes con genotipo 0/1 en la madre
- Ligada a X recesiva de novo:
    - Hija: es extraño que se produzca un 1/1 de novo, por lo que se considera que una de ellas puede venir heredada de un parental y la otra ha podido originarse de novo.
    - Hijo: al ser varón (XY), una variante recesiva ligada a X se comportaría igual que una variante ligada a X dominante de novo.
- Ligada a X recesiva heredada:
    - Hija: la variante está en uno de los parentales y la hija es 1/1. Por tanto, la aplicación buscará variantes de la madre que tengan genotipo 0/1 y variantes del padre que tengan genotipo 1/1 (si el

Para los **modelos de herencia ligados a X**, es importante recordar que los **hombres** (**XY**) estarán afectados tanto si la variante es dominante o recesiva, ya que solo poseen una copia del gen. En cambio, las **mujeres** al ser **XX**, estarán afectadas, cuando la variante sea dominante. Y cuando la variante siga un modelo recesivo podrán estar afectadas solo si ambas copias tienen la variante. Si solo

padre tiene la variante, sería afecto por ser hemicigoto), y variantes en la hija con genotipo 0|1.

- Hijo: al ser varón (XY), una variante recesiva ligada a X se comportaría igual que una variante ligada a X dominante heredada.

En variantes con modelos de herencia recesivos se puede dar que tengamos variantes en heterocigosis, es decir, que la variante solo esté en una de las copias. Dentro de los heterocigotos distinguimos dos tipos:

- Heterocigoto compuesto: se consideran dos variantes del mismo. Cada una de estas variantes pueden venir heredadas o producirse de novo. La aplicación primero realiza un filtrado a aquellos genes que tengan dos o más variantes, después filtra las variantes de los parentales y del hijo, a aquellas que tengan genotipo 0/1.

- Heterocigoto simple: se considera una variante que está presente en una copia, y que puede haberse producido de novo o ser heredada de los parentales.

2.7. Priorización

Respecto a la priorización de las variantes, se realiza mediante una serie de desplegables presentes en la segunda pestaña de la aplicación ***(¡Error! No se encuentra el origen de la referencia.)***.

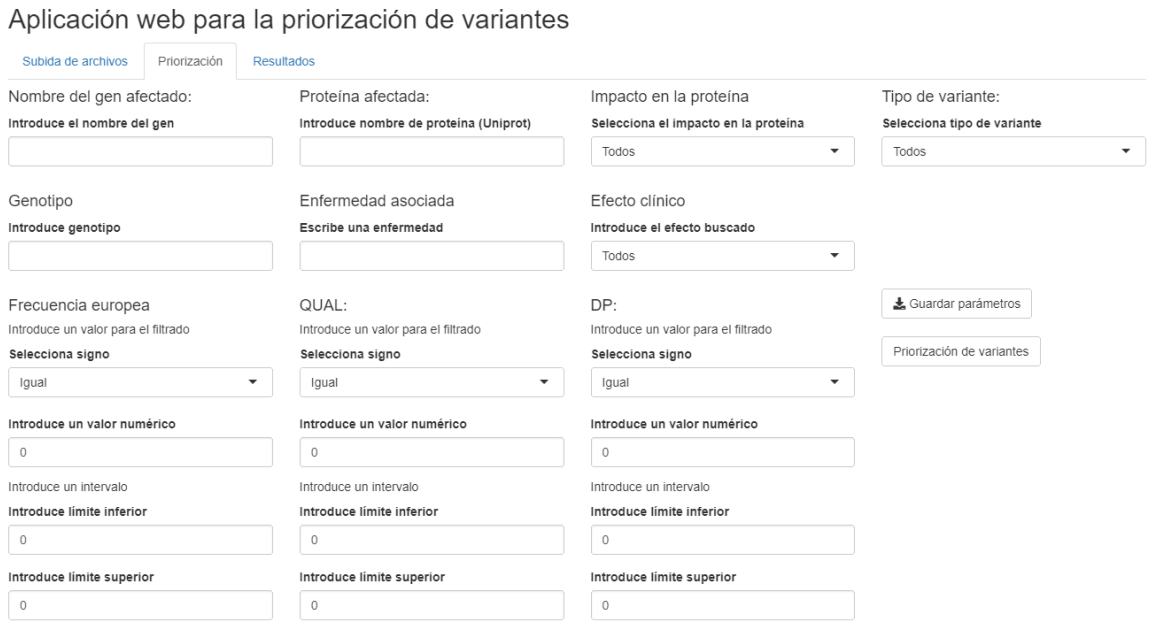


Figura 5: Parámetros para la priorización de variantes

El usuario debe ir poniendo los parámetros que le interesen, y dejando en blanco aquellos que no desee usar. Cuando estén todos los parámetros introducidos, se tendrá que hacer click en el botón “Priorización de variantes” y pasar a la pestaña de Resultados.

El botón “Guardar parámetros” permite descargar los parámetros introducidos en la aplicación en formato CSV *(**¡Error! No se encuentra el origen de la referencia.**)*.

```
"V1","V2"
"Genes",""
"Proteina",""
"Impacto en la proteína","MODERATE"
"Tipo de variante","multi-snv"
"Genotipo","0/1"
"Enfermedad asociada",""
"Efecto clínico","Malignant"
"Comparar QUAL",">= 50"
"Intervalo QUAL","0 , 0"
"Comparar DP","= 0"
"Intervalo DP","400 , 600"
"Comparar frecuencia europea",">= 0.01"
"Intervalo frecuencia europea","0 , 0"
```

Figura 6: Archivo CSV de los parámetros de priorización

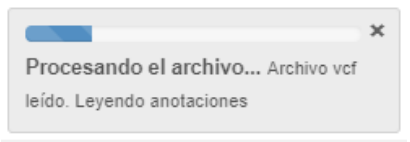
2.8. Aplicación final

La aplicación web obtenida como producto final de este proyecto, consta de tres pestañas:

2.8.1. Pestaña 1: “Subida de archivos”

Esta pestaña esta divida en dos bloques: caso índice y caso trío. En la parte izquierda, encontramos el cuadro para subir el archivo del caso índice.

En la parte derecha, correspondiente a los tríos se incluyen los apartados para subir cada uno de los archivos, y un pequeño formulario para escoger el modelo de herencia que deben seguir las variantes que se mostrarán en la última pestaña.



En ambos casos (índice y trío) una vez se complete la subida del archivo (“Upload complete”),

Figura 7: Barra de progreso

Figura 8: Pestaña 1 "Subida de archivos"

Vista de la primera pestaña de la aplicación web.

deberemos hacer click en el botón “Procesar archivos”, que hará que se comience el proceso de lectura y procesamiento de los VCFs, mostrando una barra de progreso *(Figura 7)* en la esquina inferior izquierda. Si se selecciona trío, será necesario seleccionar que modelo de herencia se va a aplicar *(Figura 8)*.

Una vez termine el procesamiento de los archivos subidos, tanto en el caso índice como en el caso trío, el aplicativo web mostrará un mensaje al final de la sección, informando al usuario de que el procedimiento se ha completado correctamente.

En el caso trío, cabe la posibilidad de que no existan variantes que cumplan el modelo de herencia seleccionado, si esto ocurre la aplicación mostrará un mensaje advirtiendolo al usuario.

2.8.2. Pestaña 2: “Priorización”

En la pestaña de priorización, se encuentran todos los filtros disponibles junto con una pequeña explicación de cada uno de los parámetros que aparecen mostrados en esta pestaña. Además, hay dos botones, el primero permite guardar en un archivo CSV los parámetros de priorización que se han utilizado, y un segundo botón que ejecuta la orden de priorización, cuyos resultados se

muestran en la tercera pestaña de la aplicación web *(**¡Error! No se encuentra el origen de la referencia.**)*.

El usario tendrá que introducir los parámetros a usar para hacer la priorización, es importante recalcar que no es necesario rellenar todos los campos. Una vez introcidos, al hacer click en “Priorización de variantes”, el aplicativo ejecutará la orden y se mostrarán los resultados en la pestaña siguiente.

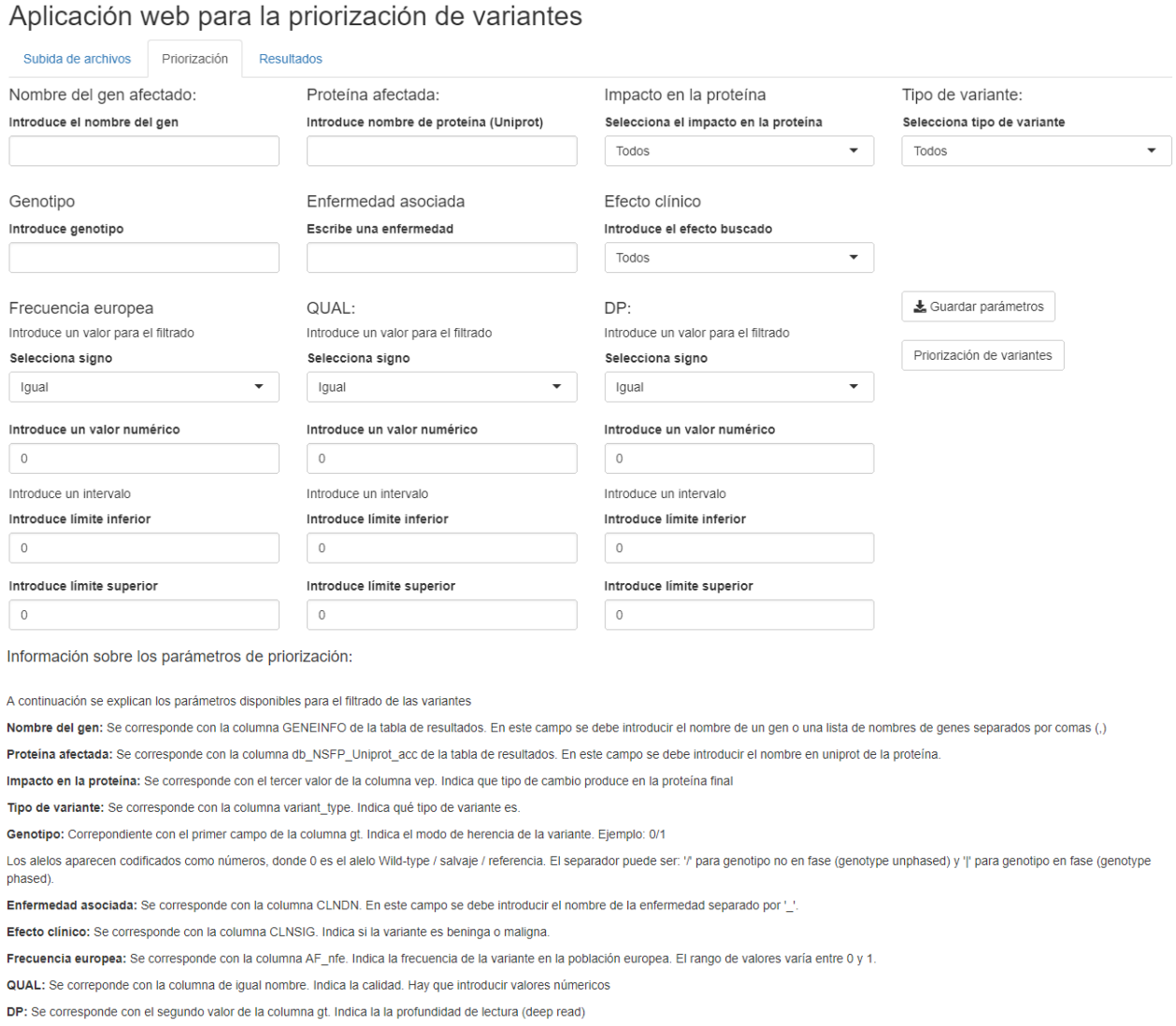


Figura 9: Pestaña "Priorización"

Vista de la segunda pestaña de la aplicación web diseñada.

2.8.3. Pestaña 3: "Resultados"

La tabla de resultados obtenida del proceso de priorización contendrá las siguientes columnas:

- CHROM
- POS
- ID
- REF
- ALT
- QUAL
- DP
- genotype
- gene
- transcript
- impact
- consequence
- variant_type
- cDNA
- exon
- intron
- AF_nfe
- dbNSFP_Uniprot_acc
- CLNSIG
- CLNDN
- CLNDISDB
- n_alt_alleles
- dbNSFP_MutationTaster_pred
- dbNSFP_Polyphen2_HDIV_pred
- dbNSFP_Polyphen2_HVAR_pred
- dbNSFP_SIFT_pred
- dbNSFP_LRT_pred

Acerca de las columnas arriba especificadas, en la aplicación se incluye una breve explicación sobre cada una de ellas.

En esta pestaña se muestran los resultados obtenidos después de todo el proceso, cuando se trabaja con un caso índice la aplicación mostrará una única tabla *(Figura 10).*

Resultado de la priorización para caso Índice

Show 10 entries

	CHROM	POS	ID	REF	ALT	QUAL	DP	genotype	gene	transcript	impact	consequence	variant_type
1	22	18300775	rs61744842	G	A	50	502	0|1	MICAL3	NM_015241.2	MODERATE	missense_variant	snv
2	22	18304784	rs5992113	A	C	50	712	0|1	MICAL3	NM_015241.2	MODIFIER	intron_variant	snv
3	22	18304821	rs45514595	G	A	50	682	0|1	MICAL3	NM_015241.2	LOW	synonymous_variant	snv
4	22	18304891	rs45544141	C	T	50	710	0|1	MICAL3	NM_015241.2	MODERATE	missense_variant	snv
5	22	18304978	rs45572336	G	A	50	758	0|1	MICAL3	NM_015241.2	MODIFIER	intron_variant	snv
6	22	18310439	rs61744286	G	A	50	599	0|1	MICAL3	NM_015241.2	MODERATE	missense_variant	snv
7	22	18348823	rs2277832	G	A	50	657	0|1	MICAL3	NM_015241.2	MODIFIER	intron_variant	snv
8	22	18566288	rs12484657;95877	C	G	50	805	0|1	PEX26	NM_001127649.2	MODERATE	missense_variant	snv
9	22	19028782	rs73157286	C	T	50	657	0|1	DGCR2	NM_005137.2	LOW	synonymous_variant	snv
10	22	19036172	rs73157291	A	G	50	632	1|0	DGCR11	NR_024157.1	MODIFIER	intron_variant	snv

Figura 10: Ejemplo de resultado para caso Índice

En esta figura, podemos ver un ejemplo de la tabla de resultados para un caso índice. Usando los parámetros Frecuencia europea menor o igual que 0.15 y tipo de variante igual a SNV.
No se muestra toda la tabla.

Mientras que cuando trabajamos con trío, se muestran tres tablas, una por cada miembro de la familia *(*Figura 11Figura 12 *y* Figura 13*).*

Resultado de la priorización para caso Trio - Madre

Show 10 entries

	CHROM	POS	ID	REF	ALT	QUAL	DP	genotype	gene	transcript	impact	consequence	variant_type
1	22	17589246	rs879576;340592	G	A	50	.	0/1	IL17RA	NM_014339.6	LOW	synonymous_variant	snv
2	22	17590180	rs41323645;340606	G	A	50	.	0/1	IL17RA	NM_014339.6	MODERATE	missense_variant	snv
3	22	18021505	rs2268776	T	C	50	.	0/1	CECR2	NM_001290047.1	MODIFIER	intron_variant	snv
4	22	18209496	rs73378798	A	G	50	.	0/1	BCL2L13	NM_001270726.1	LOW	synonymous_variant	snv
5	22	18286342	rs2241252	A	G	50	.	0/1	MICAL3	NM_015241.2	MODIFIER	intron_variant	snv
6	22	18301570	rs73876508	G	A	50	.	0/1	MICAL3	NM_015241.2	MODERATE	missense_variant	snv
7	22	18304753	rs67527329	T	C	50	.	0/1	MICAL3	NM_015241.2	MODIFIER	intron_variant	multi-snv
8	22	18304784	rs5992113	A	C	50	.	0/1	MICAL3	NM_015241.2	MODIFIER	intron_variant	snv
9	22	18304821	rs45514595	G	A	50	.	0/1	MICAL3	NM_015241.2	LOW	synonymous_variant	snv
10	22	18304891	rs45544141	C	T	50	.	0/1	MICAL3	NM_015241.2	MODERATE	missense_variant	snv

Showing 1 to 10 of 124 entries

Figura 11: : Tabla de resultados para Trío – Madre

Resultado de un caso Trío, modelo de herencia aplicado autosómica dominante heredada, parámetros de priorización usados: frecuencia europea menor o igual que 0.2.

Resultado de la priorización para caso Trio - Padre

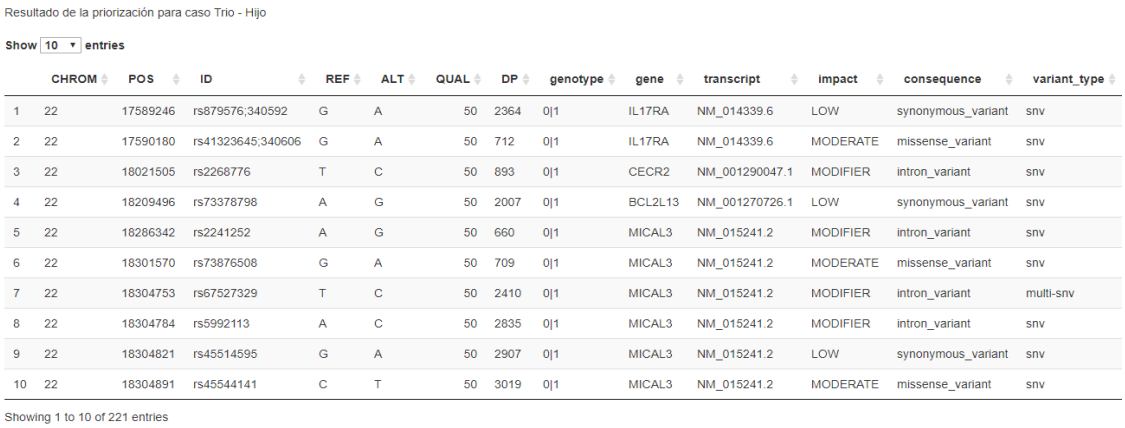
Show 10 entries

	CHROM	POS	ID	REF	ALT	QUAL	DP	genotype	gene	transcript	impact	consequence	variant_type
1	22	18072333	rs34661208	A	ACTT	50	.	0/1	LOC101929372	NM_001288707.1	MODIFIER	downstream_gene_variant	mixed
2	22	18918760	rs67625420	T	C	50	.	0/1	PRODH	NM_016335.4	MODIFIER	intron_variant	snv
3	22	19026332	rs73384823	A	G	50	.	0/1	DGCR2	NM_005137.2	MODIFIER	3_prime_UTR_variant	snv
4	22	19026356	rs112784648	G	A	50	.	0/1	DGCR2	NM_005137.2	MODIFIER	3_prime_UTR_variant	snv
5	22	19124865	rs17743887	C	T	50	.	0/1	DGCR14	NM_022719.2	MODERATE	missense_variant	multi-snv
6	22	19132062	rs113904207	G	A	50	.	0/1	DGCR14	NM_022719.2	MODIFIER	downstream_gene_variant	snv
7	22	19165216	rs2854642	G	A	50	.	0/1	LINC01311	NR_103767.1	MODIFIER	intron_variant	multi-snv
8	22	19165478	rs2070255;159908	A	G	50	.	0/1	SLC25A1	NM_001256534.1	LOW	synonymous_variant	snv
9	22	19167801	rs2793062	C	T	50	.	0/1	SLC25A1	NM_001256534.1	MODIFIER	upstream_gene_variant	multi-snv
10	22	19183787	rs1633399	A	G	50	.	0/1	CLTCL1	NM_007098.3	MODERATE	missense_variant	snv

Showing 1 to 10 of 119 entries

Figura 12: : Tabla de resultados para Trío – Padre

Resultado de un caso Trío, modelo de herencia aplicado autosómica dominante heredada, parámetros de priorización usados: frecuencia europea menor o igual que 0.2



Resultado de la priorización para caso Trio - Hijo

Show 10 entries

	CHROM	POS	ID	REF	ALT	QUAL	DP	genotype	gene	transcript	impact	consequence	variant_type
1	22	17589246	rs879576;340592	G	A	50	2364	0|1	IL17RA	NM_014339.6	LOW	synonymous_variant	snv
2	22	17590180	rs41323645;340606	G	A	50	712	0|1	IL17RA	NM_014339.6	MODERATE	missense_variant	snv
3	22	18021505	rs2268776	T	C	50	893	0|1	CECR2	NM_001290047.1	MODIFIER	intron_variant	snv
4	22	18209496	rs73378798	A	G	50	2007	0|1	BCL2L13	NM_001270726.1	LOW	synonymous_variant	snv
5	22	18286342	rs2241252	A	G	50	660	0|1	MICAL3	NM_015241.2	MODIFIER	intron_variant	snv
6	22	18301570	rs73876508	G	A	50	709	0|1	MICAL3	NM_015241.2	MODERATE	missense_variant	snv
7	22	18304753	rs67527329	T	C	50	2410	0|1	MICAL3	NM_015241.2	MODIFIER	intron_variant	multi-snv
8	22	18304784	rs5992113	A	C	50	2835	0|1	MICAL3	NM_015241.2	MODIFIER	intron_variant	snv
9	22	18304821	rs45514595	G	A	50	2907	0|1	MICAL3	NM_015241.2	LOW	synonymous_variant	snv
10	22	18304891	rs45544141	C	T	50	3019	0|1	MICAL3	NM_015241.2	MODERATE	missense_variant	snv

Showing 1 to 10 of 221 entries

Figura 13: : Tabla de resultados para Trío – Hijo

Resultado de un caso Trío, modelo de herencia aplicado autosómica dominante heredada, parámetros de priorización usados: frecuencia europea menor o igual que 0.2

Cada una de las tablas puede ser exportado por separado mediante los botones que tenemos en la parte superior derecha de la pestaña *(*Figura 14***¡Error! No se e***

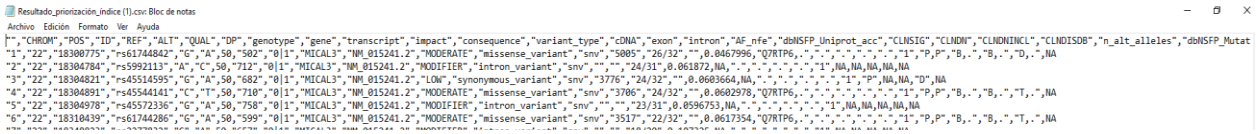
Figura 14: Botones para la exportación de resultados

ncuentra el origen de la referencia.*)*.

Cuando se hace click en los botones para exportar los resultados , el sistema nos devuelve un archivo CSV separado por comas con la tabla (Figura 15*)*.

2.9. Ejemplos de priorización.

Para estos ejemplos de priorización, usaremos el archivo de caso índice del cromosoma 22. Usaremos la aplicación para resaltar la importancia de la priorización adecuada de las variantes. Para ello, realizaremos 4 pruebas:



Resultado_priorización_índice (1).csv: Bloc de notas

Archivo Edición Formato Ver Ayuda

```
","CHROM","POS","ID","REF","ALT","QUAL","DP","genotype","gene","transcript","impact","consequence","variant_type","cDNA","exon","intron","AF_nfe","dbNSFP_Uniprot_acc","CLNSIG","CLNDN","CLNDNINCL","CLNDISDB","n_alt_alleles","dbNSFP_Mutat
"1","22","18300775","rs61744842","G","A",50,"592","0|1","MICAL3","NM_015241.2","MODERATE","missense_variant","snv","5005","26/32","",0.0467996,"Q7RTP6",".",".",".",".",".",".","1","P,P","B,.","B,.","D,.",NA
"2","22","18304784","rs5992113","A","C",50,"712","0|1","MICAL3","NM_015241.2","MODIFIER","intron_variant","snv","","","24/31",0.061872,NA,".",".",".",".",".",".","1",NA,NA,NA,NA,NA
"3","22","18304821","rs45514595","G","A",50,"682","0|1","MICAL3","NM_015241.2","LOW","synonymous_variant","snv","3776","24/32","",0.0603664,NA,".",".",".",".",".",".","1","P",NA,NA,"D",NA
"4","22","18304891","rs45544141","C","T",50,"710","0|1","MICAL3","NM_015241.2","MODERATE","missense_variant","snv","3706","24/32","",0.0602978,"Q7RTP6",".",".",".",".",".",".","1","P,P","B,.","B,.","T,.",NA
"5","22","18304978","rs45572336","G","A",50,"758","0|1","MICAL3","NM_015241.2","MODIFIER","intron_variant","snv","","","23/31",0.0596753,NA,".",".",".",".",".",".","1",NA,NA,NA,NA,NA
"6","22","18310439","rs61744286","G","A",50,"599","0|1","MICAL3","NM_015241.2","MODERATE","missense_variant","snv","3517","22/32","",0.0617354,"Q7RTP6",".",".",".",".",".",".","1","P,P","B,.","B,.","T,.",NA
```

Figura 15: Resultado de la exportación de una tabla de resultados

Exportación del resultado de la priorización de un caso índice. Parámetros de priorización usados: Tipo de variante igual a SNV y frecuencia europea menor o igual a 0.2.
No se muestran todas las columnas, únicamente las que entran en pantalla.

- Primera prueba: después de importar el archivo, y sin introducir ningún parámetro de priorización, se mirará el número de variantes obtenidas.
- Segunda prueba: se introducen dos parámetros de priorización . Los parámetros usados son:
    - Tipo de variante: SNV
    - Frecuencia europea ≤ 0.35
- Tercera prueba: se introducen los dos anteriores más dos parámetros de priorización nuevos.
    - DP entre 600 y 900
    - Impacto en la proteína: moderado
- Cuarta prueba: se introducen los cuatro anteriores más un parámetro de priorización nuevo.
    - Nombre del gen: PRR14L

El resultado se presenta en la
Tabla 1:

Tabla 1: Resultados de la comparativa de direferentes conjuntos de parámetros de priorización. Caso índice

Primera prueba: Número de variantes sin priorizar	44990
Segunda prueba: Número de variantes después de priorizar con dos parámetros	289
Tercera prueba: Número de variantes después de priorizar con cuatro parámetros	39
Tercera prueba: Número de variantes después de priorizar con cinco parámetros	3

Como podemos ver, aplicando correctamente diferentes parámetros de priorización podemos llegar a reducir a unas pocas variantes.

3. Conclusiones

Durante el desarrollo de este trabajo, se ha aprendido las etapas fundamentales del diseño y creación de una aplicación web, con las que se no estaba nada familiarizada. Se ha aprendido a programar aplicaciones con el paquete Shiny,

que hasta el momento solo se había trabajado con aplicaciones hechas por terceras personas.

En cuanto al cumplimiento de objetivos se consideran logrados, la aplicación obtenida es completamente funcional, siendo capaz de aplicar los modelos de herencia correctamente y realizar la priorización de una forma sencilla para el usuario y rápida.

La planificación en general ha sido adecuada, aunque la tarea de anotación de los archivos inicialmente no estaba definida como tal, y debido a ello se retrasó un poco el diseño del algoritmo para realizar la priorización. Este inconveniente se solucionó dedicandole más horas para compensar el retraso.

Una de las líneas para el trabajo futuro podría ser la posibilidad de incluir interaciones entre genes como por ejemplo las epistasis o las pleiotropías. Otra línea sería añadir la opción de exportar las tablas de resultados como VCF, y por último, aumentar el número de parámetros que el usario puede ajustar.

Mi valoración general del TFM es muy positiva, ya que considero que he aprendido mucho sobre todo en cuanto a informática y a programación se refiere, consolidando todo lo aprendido en el máster. Sobre la parte más biológica del trabajo, ya tenía experiencia en cuanto a entender qué efectos podría tener las variaciones en el genoma sobre la expresión génica, siempre desde un punto de vista biológico, sin entrar en la parte más bioinformática.

4. Glosario

ADN: ácido desoxirribonucleico

Epistasis: cuando la expresión de un gen enmascara o suprime la expresión de otro.

Fenotipo: expresión del genotipo en función del ambiente donde viva el organismo

GATK: Genome Analtsis Toolkit. Aplicación para analisis de datos provenientes de tecnologías de secuenciación Next-Generation Sequencing.

Genoma: conjunto del material genético de un organismo, tipicamente en forma de ADN.

Genotipo: información genética que posee un organismo determinado, normalmente en forma de ADN.

GitHub: es una plataforma para el desarrollo colaborativo de software, que permite alogar proyectos usando el sistema de control de versiones Git.

Illumina: Empresa estadounidense dedicada a secuenciar usando Next-Generation Sequencing. Comercializa sistemas para el análisis de variaciones en el genoma y función biológica.

Indel: contracción de inserción – deleción, hace referencia a un tipo de mutaciones genéticas que consisten en pequeñas inserciones o deleciones.

NGS: Next Generation Sequencing.

Pleiotropía: cuando un cambio en un gen provoca la aparición de distintos fenotipos.

SNV: Single Nucleotide Variant o Variante de Nucleótido Único.

VCF: Variant Format Calling, archivo de texto que se usa en bioinformática para almacenar datos sobre variantes genéticas y sus anotaciones.

WES: Whole Exome Sequencing o secuenciación de exoma completo.

WGS: Whole Genome Secuencing o secuenciación de genoma completo

5. Bibliografía

[1] J. M. Zook *et al.*, «Integrating human sequence data sets provides a resource of benchmark SNP and indel genotype calls», *Nat. Biotechnol.*, vol. 32, n.º 3, pp. 246-251, mar. 2014.

[2] J. M. Zook *et al.*, «Extensive sequencing of seven human genomes to characterize benchmark reference materials», *Scientific Data*, vol. 3, p. 160025, jun. 2016.

[3] M. Lek *et al.*, «Analysis of protein-coding genetic variation in 60,706 humans», *Nature*, vol. 536, n.º 7616, pp. 285-291, ago. 2016.

[4] A. Kitts y S. Sherry, *The Single Nucleotide Polymorphism Database (dbSNP) of Nucleotide Sequence Variation*. National Center for Biotechnology Information (US), 2011.

[5] C. Dong *et al.*, «Comparison and integration of deleteriousness prediction methods for nonsynonymous SNVs in whole exome sequencing studies», *Hum Mol Genet*, vol. 24, n.º 8, pp. 2125-2137, abr. 2015.

[6] M. J. Landrum *et al.*, «ClinVar: public archive of relationships among sequence variation and human phenotype», *Nucleic Acids Res*, vol. 42, n.º D1, pp. D980-D985, ene. 2014.

[7] P. Cingolani *et al.*, «A program for annotating and predicting the effects of single nucleotide polymorphisms, SnpEff», *Fly (Austin)*, vol. 6, n.º 2, pp. 80-92, abr. 2012.

[8] W. Chang *et al.*, *shiny: Web Application Framework for R*. 2019.

[9] F. Sanger, S. Nicklen, y A. R. Coulson, «DNA sequencing with chain-terminating inhibitors», *Proc. Natl. Acad. Sci. U.S.A.*, vol. 74, n.º 12, pp. 5463-5467, dic. 1977.

[10] J. M. Prober *et al.*, «A system for rapid DNA sequencing with fluorescent chain-terminating dideoxynucleotides», *Science*, vol. 238, n.º 4825, pp. 336-341, oct. 1987.

[11] B.-S. Petersen, B. Fredrich, M. p. Hoeppner, D. Ellinghaus, y A. Franke, «Opportunities and challenges of whole-genome and -exome sequencing», *BMC Genetics*, vol. 18, n.º 1, 14 2017.

[12] E. S. Lander *et al.*, «Initial sequencing and analysis of the human genome», *Nature*, vol. 409, n.º 6822, pp. 860-921, feb. 2001.

[13] M. Margulies *et al.*, «Genome sequencing in microfabricated high-density picolitre reactors», *Nature*, vol. 437, n.º 7057, pp. 376-380, sep. 2005.

[14] D. R. Bentley *et al.*, «Accurate whole human genome sequencing using reversible terminator chemistry», *Nature*, vol. 456, n.º 7218, pp. 53-59, nov. 2008.

[15] N. Prodduturi, A. Bhagwate, J.-P. A. Kocher, y Z. Sun, «Indel sensitive and comprehensive variant/mutation detection from RNA sequencing data for precision medicine», *BMC Med Genomics*, vol. 11, n.º Suppl 3, sep. 2018.

[16] S. Salatino y V. Ramraj, «BrowseVCF: a web-based application and workflow to quickly prioritize disease-causative variants in VCF files», *Briefings in Bioinformatics*, vol. 18, n.º 5, p. 774, sep. 2017.

[17] Z. Lai *et al.*, «VarDict: a novel and versatile variant caller for next-generation sequencing in cancer research», *Nucleic Acids Res.*, vol. 44, n.º 11, p. e108, 20 2016.

[18] Y. Liu, M. Loewer, S. Aluru, y B. Schmidt, «SNVSniffer: an integrated caller for germline and somatic single-nucleotide and indel mutations», *BMC Syst Biol*, vol. 10 Suppl 2, p. 47, 01 2016.

[19] C. Kockan *et al.*, «SiNVICT: ultra-sensitive detection of single nucleotide variants and indels in circulating tumour DNA», *Bioinformatics*, vol. 33, n.º 1, pp. 26-34, 01 2017.

[20] «GEMINI: Integrative Exploration of Genetic Variation and Genome Annotations». [En línea]. Disponible en: https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003153. [Accedido: 04-mar-2019].

[21] J. Delafontaine, A. Masselot, R. Liechti, D. Kuznetsov, I. Xenarios, y S. Pradervand, «Varapp: A reactive web-application for variants filtering», *bioRxiv*, jun. 2016.

[22] R. J. Schmidt, A. Macleay, y L. P. Le, «VarGrouper – a Bioinformatic Tool for Local Haplotyping of Deletion-Insertion Variants from Next-generation Sequencing Data Post Variant Calling», *The Journal of Molecular Diagnostics*, feb. 2019.

[23] «GATK | Home». [En línea]. Disponible en: https://software.broadinstitute.org/gatk/. [Accedido: 01-jun-2019].

[24] B. J. Knaus y N. J. Grünwald, «vcfr: a package to manipulate and visualize variant call format data in R», *Molecular Ecology Resources*, vol. 17, n.º 1, pp. 44-53, 2017.

[25] Y. Xie *et al.*, *DT: A Wrapper of the JavaScript Library «DataTables»*. 2019.

[26] «Shiny - Reactivity - An overview». [En línea]. Disponible en: https://shiny.rstudio.com/articles/reactivity-overview.html. [Accedido: 30-may-2019].