Escuela Superior
de Ingeniería y Tecnología
Universidad de La Laguna

Trabajo de Fin de Grado

Grado en Ingeniería Informática

Herramienta de Gestión de Grafos de Variación para Bioinformática

Variation Graph Management Tool for Bioinformatics

La Laguna, 8 de septiembre de 2021

D. **Marcos Colebrook Santamaría**, con N.I.F. 43.787.808-V, profesor Titular de Universidad adscrito al Departamento de Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como tutor

D. **José L. Roda García**, con N.I.F. 43.356.123-L profesor Titular de Universidad adscrito al Departamento de Nombre del Ingeniería Informática y de Sistemas de la Universidad de La Laguna, como cotutor

C E R T I F I C A (N)

Que la presente memoria titulada:

"Herramienta de Gestión de Grafos de Variación para Bioinformática"

ha sido realizada bajo su dirección por D. **Elliott Dorta Ramos**,

con N.I.F. 54.108.407-X.

Y para que así conste, en cumplimiento de la legislación vigente y a los efectos oportunos firman la presente en La Laguna a 8 de septiembre de 2021

Agradecimientos

Me gustaría agradecer a los directores del proyecto el Dr. Marcos Colebrook y Dr. José Luis Roda, por aportarme su experiencia, por mostrarme una rama de estudio como la bioinformática, por la motivación, por la cercanía, por sus conocimientos, gracias.

Sin dudar a mi familia esos que siempre estuvieron ahí y sacrificaron tanto de su vida para que yo tuviera una mejor que ellos, no sabré nunca como agradecerles este gesto constante y la fe e ilusión para que yo llegara aquí.

A mis referentes en cuanto a superación, nobleza e incondicionalidad Ayoze, Jonay y Alejandro por su puesto a su santa madre Mini que sin esas broncas yo no hubiese llegado ni a sacar una asignatura de esta carrera, aunque también fueron necesarios los cariños de Marinieves.

A una de las mejores cosas que me han pasado en la vida, mis "brothers" a todos ellos que me han sumado Alvaro, Jeus, Xovo, Aurich, Adrián, Xavi, Fran, Macho, Carlitos, simplemente ¡siempre juntos, siempre fuertes!

A mis mecenas por su puesto don Israel y doña Carmen aquellos que patrocinaron mis últimos años de carrera y me apoyaron para que finalizar mis estudios, en cuanto a Tato no lo tengo tan claro, pero definitivamente gracias por todas esas veces que me complicaste las tardes.

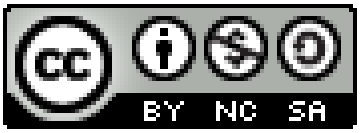
A los que me enseñaron la definición absoluta y sin indirectas de la palabra locura Melissa, Fela, José, Patri, Dessi y Guaci deberían ir mirando psiquiatras, pero mientras sigamos siendo amigos.

Por supuesto a la que me enseñó a que lo bueno se encuentra cuando menos buscas, a la que me hizo entender porque a veces los huracanes tienes nombre de mujer, a la que me levanto más de una vez, Adara mil gracias.

A mis abuelos me enseñaron tanto o igual que la carrera.

En resumidas cuentas, a todos los anteriores los amo gracias por hacerme feliz y ser parte de su vida.

Licencia



Resumen

El objetivo de este proyecto de Bioinformática es el estudio de herramientas para la representación de las variaciones del genoma, así como el análisis y funciones de estas para obtener una formación en dicho ámbito.

Para cubrir este objetivo se han desarrollado tres cuadernos el primer cuaderno arrancará la herramienta motor y común para todos los cuadernos llamada VG, la cual es una herramienta de generación y gestión de grafos de variación.

Por otro lado, se abordará como interpretar estos datos y que sean lo más legibles posible para el usuario. Para ello se han preparado dos cuadernos más con dos herramientas de visualización de grafos de variación diferentes que son Sequence Tube Maps, la cual usa los gráficos de tipo tubería simulando las líneas de metro; y por otro lado MoMI-G, que usa tres tipos de representación: una circular, otra que utiliza Sequence Tube Map dentro de la propia aplicación y, para finalizar, una representación directa en nucleótidos basada también en Sequence Tube Map.

Palabras clave: Bioinformática, secuenciación genómica, grafos de variación, Jupyter, VG, Sequence Tube Maps, MoMI-G

Abstract

The aim of this Bioinformatics project is the study of tools for the representation of the variations of the genome as well as the analysis and functions of the same in order to obtain training in this field.

To cover this objective, three notebooks have been developed. The first notebook will start with the motor tool common to all the notebooks called VG, which is a tool for the generation and management of variation graphs.

On the other hand, it will deal with how to interpret these data and make them as readable as possible for the user. For this purpose, two more notebooks have been prepared with two different variation graph visualisation tools: Sequence Tube Maps, which uses pipe-type graphs simulating metro lines; and MoMI-G, which uses three types of representation: a circular one, another one that uses Sequence Tube Map within the application itself and, finally, a direct representation in nucleotides also based on Sequence Tube Map.

Keywords: Bioinformatics, genomic sequencing, variation graphs, Jupyter, VG, Sequence Tube Maps, MoMI-G.

Índice general

Índice de figuras

Índice de tablas

Capítulo 1 Introducción

1.1 ¿Qué es la Bioinformática?

Con un gran número de genomas procariotas y eucariotas completamente secuenciados, el acceso a la información genómica y su síntesis para el descubrimiento de nuevos conocimientos se han convertido en temas centrales de la investigación biológica moderna. La extracción de información genómica requiere el uso de sofisticadas herramientas computacionales.

Por lo tanto, se vuelve imperativo para la nueva generación de biólogos iniciar y familiarizarse con un campo de estudio que se preocupa por el almacenamiento, organización e indexación cuidadosa de la información para abordar los nuevos desafíos en la era genómica. La ciencia de la computación se ha aplicado a la biología para producir un campo que se llama bioinformática.

La Bioinformática se ocupa de las herramientas computacionales de última generación disponibles para resolver problemas de investigación biológica.

De este modo, el término bioinformática [1] fue acuñado por Paulien Hogeweg y Ben Hesper para describir "*el estudio de los procesos informáticos en sistemas bióticos*". Encontró un uso temprano cuando comenzaron a compartirse los primeros datos de secuencias biológicas. La bioinformática es un campo interdisciplinario que desarrolla métodos y herramientas de software para comprender datos biológicos.

El desarrollo de la bioinformática como campo es el resultado de los avances tanto en biología molecular como en informática durante los últimos 30 a 40 años. Como campo interdisciplinario de la ciencia, la bioinformática combina biología, ciencias de la computación, ingeniería de la información, matemáticas y estadística para analizar e interpretar datos biológicos.

Las áreas clave de la bioinformática incluyen bases de datos biológicas, alineación de secuencias, predicción de genes y promotores, filogenia molecular, variaciones estructurales, variaciones en las secuencias genómicas, bioinformática estructural, genómica y proteómica.

1.2 Diferencia entre Bioinformática y Biología Computacional

La bioinformática se limita al análisis secuencial, estructural y funcional de genes y genomas y sus productos correspondientes y, a menudo, se considera biología molecular computacional. Sin embargo, la biología computacional abarca todas las áreas biológicas que involucran computación como el desarrollo de los algoritmos y métodos estadísticos para los análisis biológicos.

La bioinformática como el desarrollo y la aplicación de herramientas computacionales en la gestión de todo tipo de datos biológicos, mientras que la biología computacional se limita más al desarrollo teórico de algoritmos utilizados para la bioinformática.

Aunque en conceptos son diferentes los términos se usan indistintamente.

Figura 1: Pantalla con visualización de datos

1.3 Representación del genoma

El genoma se refiere a todo el ADN presente en un organismo. El ADN es el "modelo genético" que determina la composición genotípica de cada organismo. En su forma más simple, el ADN consta de dos cadenas de nucleótidos o bases (abreviadas A, C, G y T), enrolladas una alrededor de la otra.

Las bases que componen el ADN tienen capacidades de unión específicas: A siempre se une a T y C siempre se une a G. Estas capacidades de unión son útiles para que los científicos las comprendan, ya que, si se determina la secuencia de nucleótidos de una hebra de ADN, la unión complementaria permite la secuencia de otra hebra por deducir.

En el caso de los humanos, el ADN está organizado en 23 unidades estructurales llamadas cromosomas. Cada cromosoma consta de espirales compactadas de ADN. Si bien gran parte de este ADN no tiene una función conocida (estos tramos de ADN se denominan convenientemente ADN espaciador o ADN basura), una parte significativa del ADN codifica genes. Cada gen aporta la información necesaria para producir una proteína, que se encarga de realizar funciones celulares. El complemento de proteínas en un organismo es muy importante, y las enfermedades a menudo se manifiestan cuando una proteína no funciona correctamente.

No podemos hablar de la representación del genoma sin nombrar la **secuenciación de ADN**, que es el proceso que determina la secuencia de bases de los nucleótidos (As, Ts, Cs y Gs), mencionadas anteriormente de un fragmento de ADN. Hoy esta tarea es sencilla pero no siempre ha sido así, secuenciar un genoma completo (todo el ADN de un organismo) sigue siendo una tarea compleja.

El proceso requiere romper el ADN del genoma en muchos pedazos más pequeños, secuenciar dichos pedazos y ensamblar las secuencias en una única y larga **"secuencia consenso"**. Sin embargo, gracias a nuevos métodos que se han desarrollado en las últimas dos décadas, ahora secuenciar un genoma es mucho más rápido y menos costoso.

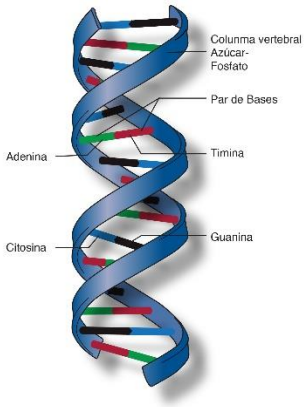


Figura 1.2: Hélice molécula de ADN

1.3.1 Secuenciación de Sanger

El método de secuenciación de Sanger [2] fue de los primeros métodos de secuenciación que se practicaron consintiendo en rutinariamente secuenciar regiones de ADN basándose en la polimerización del ADN y el uso de dideoxinucleótidos que sirven como terminadores de la reacción. En la actualidad la reacción de secuenciación se basa en una modificación de la PCR con dideoxinucleótidos marcados con fluoróforos y se resuelve mediante una electroforesis capilar.

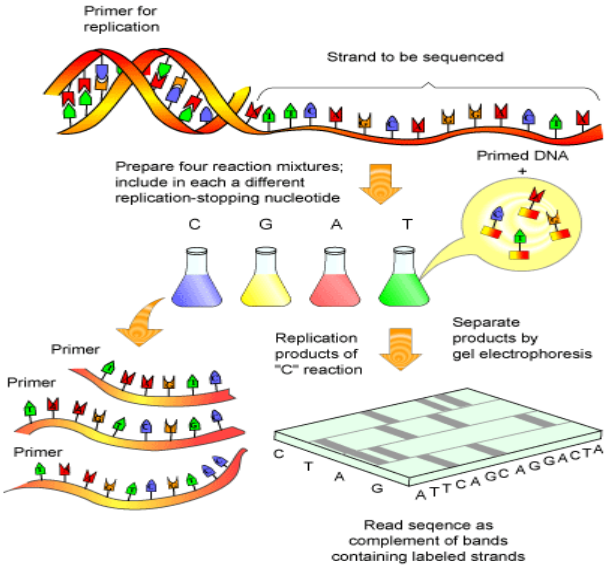


Figura 1.3: Secuenciación de Sanger

1.3.2 Secuenciación de nueva generación

El conjunto más reciente de tecnologías de secuenciación de ADN se denominan secuenciación de nueva generación [3], cada una de estas técnicas utilizan diferentes tecnologías, pero la mayoría comparten un conjunto común de características que las distinguen de la secuenciación de Sanger:

Altamente paralelas: ocurren muchas reacciones de secuenciación al mismo tiempo.

Microescala: las reacciones son diminutas y se pueden hacer muchas a la vez en un chip.

Rápidas: puesto que las reacciones se realizan en paralelo, los resultados están listos mucho más rápido.

Bajo costo: secuenciar un genoma es más barato que con la secuenciación de Sanger.

Longitudes más cortas: típicamente, las lecturas se obtienen con fragmentos de entre

505050 -700700700 nucleótidos de longitud.

En resumen, la secuenciación de nueva generación es como realizar un número muy grande de pequeñas reacciones de secuenciación de Sanger en paralelo, logrando gracias a esta paralelización y a la pequeña escala secuenciar grandes cantidades de ADN de forma mucho más rápida y barata.

1.4 JupyterLab

Jupyter [4]Notebook es una aplicación cliente-servidor lanzada en 2015 por la organización sin ánimo de lucro Proyecto Jupyter. Permite crear y compartir documentos web en formato JSON que siguen un esquema versionado y una lista ordenada de celdas de entrada y de salida. Estas celdas albergan, entre otras cosas, código, texto (en formato *Markdown*), fórmulas matemáticas y ecuaciones, o también contenido multimedia (*Rich Media*). El programa se ejecuta desde la aplicación web cliente que funciona en cualquier navegador estándar. El requisito previo es instalar y ejecutar en el sistema el servidor Jupyter Notebook. Los documentos creados en Jupyter pueden exportarse, entre otros formatos, a HTML, PDF, Markdown o Python y también pueden compartirse con otros usuarios por correo electrónico, utilizando Dropbox o GitHub o mediante el visor integrado de Jupyter Notebook.

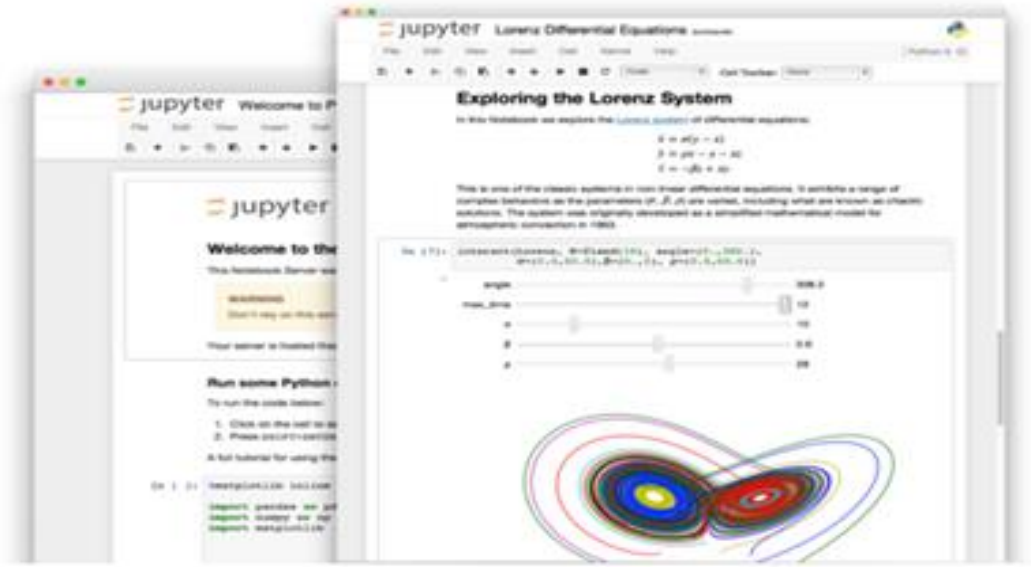


Figura 1.4: Imagen JupyterNotebook

1.4.1 Algunas características de Jupyter

- De fácil instalación gracias a estar presente en la Suite Anaconda Distribution.
- Posee una avanzada interfaz web que permite combinar código fuente, textos, fórmulas, figuras y multimedia en un solo documento.
- La integración de diversos tipos de información nos permite dar explicaciones más adecuadas de nuestros programas o de los conceptos que estemos aprendiendo.
- Permite el acceso desde cualquier lugar sin necesidad de instalación de otros servicios, ya que funciona como cliente servidor. De igual manera, Se puede ejecutar en un escritorio local o en servidor remoto.
- Aunque el lenguaje de programación fundamental en Jupyter Notebook es Python, esta aplicación también es compatible con más de 40 lenguajes, entre los que destacan R, Julia y Scala.
- Permite el intercambio de documentos de Jupyter a través de servicios de terceros.
- Podemos ejecutar y visualizar imágenes, videos, LaTeX y JavaScript, además de manipular los resultados de estos en tiempo real.
- Cuenta con un administrador de documentos avanzado, que permite visualizar los archivos compatibles con Jupyter Notebook que estén alojados en nuestro equipo.
- Los documentos realizados en Jupyter Notebook se pueden exportar a diferentes formatos estáticos incluyendo HTML, reStructeredText, LaTeX, PDF y presentaciones de diapositivas.
- Es compatible con nbviewer el cual permite portar nuestros documentos de Jupyter Notebook a la nube como una página web estática, la cual podrá ser visualizada por cualquiera sin necesidad de instalar el Jupyter Notebook.

1.5 Docker

Docker [5] es una herramienta que pretende crear contenedores ligeras y portables para las que las aplicaciones software puedan ejecutarse en cualquier máquina con Docker instalado independiente del sistema operativo de la misma, facilitando así los despliegues.

Normalmente, cuando queremos hacer cualquier tipo de instalación de software, algunas veces necesitamos tener una serie de requisitos para que este software funcione de manera correcta, por ejemplo, una versión de Java específica. Docker nos permite meter todas esas dependencias necesarias para la ejecución en un contenedor, y con ello conseguimos que la aplicación corra en cualquier máquina que disponga de Docker instalado.

Figura 1.5: Logo de Docker

1.6 Objetivos y requisitos

El objetivo principal de este Trabajo de Fin de Grado es el estudio de herramientas para la representación de las variaciones del genoma. Con estas herramientas se podrán evaluar sus características y funciones para obtener una formación en dichas herramientas.

Para lograr esos objetivos, se propone desarrollar un total de tres cuadernos de Jupyter en los que se desplieguen tres aplicaciones.

El primer cuaderno desplegará VG, que es una herramienta que genera grafos de variación, así como algunas de sus funciones básicas para que el usuario aprenda a usarla;

El segundo cuaderno será un visualizador de genoma a través de grafos de tuberías llamada Sequence Tube Maps que utiliza la herramienta del primer cuaderno como motor para generar los grafos de variación necesarios para mostrar la información; en el tercer cuaderno se muestra otra aplicación de visualización MoMI-G que utiliza VG de igual

manera.

Los datos con los que se trabajará serán datos de prueba que traen las propias aplicaciones en cada una de las herramientas. Son diferentes las secuencias generadas, y esto nos dará una perspectiva de los diferentes modelos que se pueden llegar a generar. Para obtener estos datos hay que tener instalado el ecosistema Jupyter en un ordenador o servidor con sistema UNIX para poder abrir los cuadernos y, a través de estos, poder obtener esos datos y empezar el estudio.

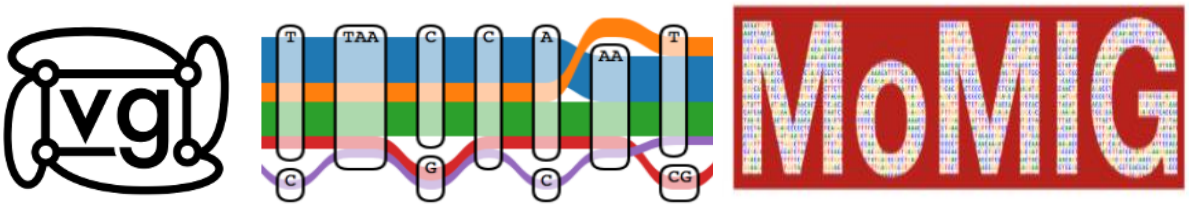


Figura 1.6: Imagen Jupyter Notebook

Capítulo 2 Estado del Arte

En el desarrollo de cualquier proyecto de investigación en el campo de la bioinformática, se depende en gran parte de los datos que se obtiene de las secuenciaciones. Dichos datos se pueden representar de varias maneras, pero hace unos años se creó un concepto llamado grafo de variación, un grafo simple y universal para sistemas de referencia pangenómicos.

Obviamente, antes de que este tipo de grafo se desarrollara ya había herramientas que representaban estos datos, pero no de la manera que lo hace la herramienta VG [6], ya que combina tres tipos de elementos en una estructura de datos pangenómica.

Estos grafos nos permiten expresar el formato VCF [9] del que es dependiente la herramienta VG para poder generar estos grafos de variación. Por tanto, estos grafos no solo son más intuitivo a la hora de leer o entender la información genómica, sino que también nos representan el aspecto que anteriormente no era posible representar.

Además, estos grafos de variación disponen de varias maneras de ser representados normalmente usando la herramienta Graphviz [10], pero en este Trabajo de Fin de Grado nos centraremos en usar las herramientas Sequence Tube Maps [7] y MoMI-G [8], para dar una alternativa a la representación de la información, ya que nos dan una manera más elegante, clara e intuitiva de representar los datos generados por los grafos de variación aparte de poder interactuar y poder seleccionar filtros que nos ayudan observar de la secuencia , cosa que Graphviz no permite hacer.

Por tanto, teniendo en cuenta lo mencionado anteriormente nos hemos centrado en el estudio de herramientas que permitan una visualización de estos grafos que nos ayuden a entender la información genética de una manera más simple, alejándonos de la comparativas con otros grafos u otras maneras de representar la información genética.

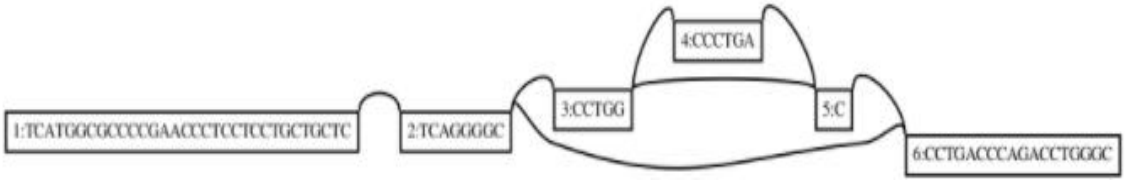


Figura 2.1: Grafos de variación.

Capítulo 3 Formato VCF y herramienta VG

3.1 Proceso de generación del archivo VCF desde FASTQ

Este apartado explicará cómo se crea un formato VCF, desde el primer momento donde la secuenciación se vuelca en el fichero de formato FASTQ hasta llegar al mismo.

- **FASTQ** [11]**:** Como bien se comentaba con anterioridad se parte de un fichero con formato FASTQ, el cual es un fichero de texto plano que contiene los datos crudos obtenidos por el secuenciador, siendo el volcado de estos datos diferente dependiendo del secuenciador que se use. Este tipo de fichero es el estándar que reconocen las herramientas bioinformáticas que trataran su contenido para obtener una alineación. Cabe mencionar que hay dos tipos secuenciación **"single end"** que son más fáciles de alinear con el genoma de referencia y **"pair end"** con una mayor dificultad de alineación. Se suele optar por la segunda opción por los datos de alta calidad que se obtienen.
- **BAM** [12]**:** Comienza el procesamiento previo de los datos de secuencia obtenidos del FASTQ, y en esta etapa realiza la alineación con un genoma de referencia, así como algunas operaciones de limpieza de datos para corregir los sesgos técnicos y hacer que los datos sean adecuados para el análisis. En este procesamiento va avanzando por las etapas de Trimming (herramienta de recorte), Mapping (mapeo de las lecturas), Mark duplicates (localiza y etiqueta las lecturas duplicadas), Recalibración (detecta errores en una etapa previa que se pueden dar al obtener la calidad de la base).
- **VARIATION CALLING** [13]**:** Hasta ahora solo hemos visto como se preprocesa la información y, a continuación, se llega al momento de las llamadas de variación donde se genera el fichero VCF. El primer proceso es el de HaplotypeCaller capaz de invocar SNPs e indels simultáneamente a través de ensamblaje *de novo* local de haplotipos en una región activa; el segundo proceso, Filtering se realiza el filtrado de manera independiente en un primer momento y luego se combinan dichos filtrados. El último proceso antes de obtener el fichero VCF es el VEP que determina el efecto de las variantes SNP, inserciones, elecciones, variantes estructurales en genes. Finalmente se obtiene el fichero VCF.

3.2 El formato VCF

El formato VCF puede entenderse como la definición de un grafo parcialmente ordenado similar a los implicados por una alineación de secuencia múltiple. Este formato ha mejorado significativamente la interoperabilidad de las herramientas de próxima generación para la alineación, visualización y variantes estructurales, por ese motivo se está empezando a pensar como el estándar a seguir. Este formato se desarrolló con la intención principal de respetar la variación genética humana, pero su uso no se limita a los genomas diploides, también se pueden utilizar en diferentes contextos por su flexibilidad y extensibilidad le permite al usuario la representación información en varios ámbitos.

El formato GFF [15] (formato de características genéticas) se ha ampliado recientemente para estandarizar el almacenamiento de información de variaciones en formato del genoma GVF, pero el problema de este formato es que no puede almacenar una cantidad alta de muestras. En cambio, VCF al ser escalable puede abarcar esas muestras de mayor tamaño.

Para la finalización de este apartado cabe a mencionar que existe una herramienta open-source llamada VCFtools [14] que sirve para analizar y manipular VCF. El primer módulo proporciona una API de Perl general y permite realizar varias operaciones en archivos VCF, incluida la validación de formato, la fusión, la comparación, la intersección, la creación de complementos y las estadísticas generales básicas. El segundo módulo consta de un ejecutable C ++ que se utiliza principalmente para analizar datos SNP en formato VCF, lo que permite al usuario estimar frecuencias de alelos, niveles de desequilibrio de ligamiento y diversas métricas de control de calidad.

3.2.1 Composición del fichero con extensión VCF

Un archivo VCF consta de una sección de encabezado y una sección de datos. El encabezado contiene un número arbitrario de líneas de metainformación, cada una de las cuales comienza con los caracteres '##', y una línea de definición de campo delimitada por TAB, que comienza con un solo carácter '#'. Las líneas de encabezado de metainformación proporcionan una descripción estandarizada de las etiquetas y anotaciones utilizadas en la sección de datos.

El uso de metainformación permite que la información almacenada en un archivo VCF se adapte al conjunto de datos en cuestión. También se puede utilizar para proporcionar información sobre los medios de creación del archivo, la fecha de creación, la versión de la secuencia de referencia, el software utilizado y cualquier otra información relevante para el historial del archivo.

La línea de definición de campo nombra ocho columnas obligatorias, correspondientes a las columnas de datos:

- CHROM: Representan el cromosoma.
- POS: Posición basada en 1 del inicio de la variante.
- ID: Identificadores únicos de la variante.

- REF: Alelo de referencia.
- ALT: Lista separada por comas de alelos alternativos que no son de referencia.
- QUAL: Puntuación de calidad en escala phred.
- FILTER: Información de filtrado del sitio.
- INFO: Lista separada por puntos y comas que expone las anotaciones adicionales por el usuario.
- FORMAT: se utiliza para definir la información contenida en cada columna de genotipo subsiguiente, que consiste en una lista de campos separados por dos puntos.

Por ejemplo, el campo FORMAT GT: GQ: DP en la cuarta entrada de datos de una puntuación de calidad en escala phred (QUAL), información de filtrado del sitio (FILTER) y una lista separada por punto y coma de anotaciones adicionales ampliables por el usuario (INFO). Además, si hay muestras en el archivo, las columnas de encabezado obligatorias van seguidas de una columna FORMAT y un número arbitrario de ID de muestra que definen las muestras incluidas en el archivo VCF.

Dentro de la especificación anteriormente mencionada VCF tiene palabras reservadas a continuación mostraremos las más significativas, pero en caso de querer ver la especificación completa del formato podrán acceder al manual en el sitio oficial.

Para la columna del campo Genotipo:

- GT: genotipo, codifica alelos como números: 0 para el alelo de referencia, 1 para el primer alelo enumerado en la columna ALT, 2 para el segundo alelo enumerado en ALT y así sucesivamente. El número de alelos sugiere ploidía de la muestra y el separador indica si los alelos están en fase ('|') o sin fase ('/') con respecto a otras líneas de datos.
- PS: conjunto de fases, indica que los alelos de genotipos con el mismo valor de PS se enumeran en el mismo orden.
- DP: lee la profundidad en esta posición.
- GL: probabilidades de genotipo para todos los posibles genotipos dado el conjunto de alelos definidos en los campos REF y ALT.
- GQ: calidad del genotipo, probabilidad de que la llamada del genotipo sea incorrecta con la condición de que el sitio sea una variante. Tenga en cuenta que la columna CUAL proporciona un puntaje de calidad general para la afirmación hecha en ALT de que el sitio es variante o no varía.

Para la columna de campo INFO:

- DB: membresía dbSNP.
- H3: membresía en HapMap3.
- VALIDATED: validado por experimento de seguimiento.

- AN: número total de alelos en genotipos llamados.
- AC: recuento de alelos en genotipos, para cada alelo ALT, en el mismo orden en el que se enumeran.
- SVTYPE: tipo de variante estructural (DEL para deleción, DUP para duplicación, INV para inversión, etc. cómo se describe en la especificación).
- END: posición final de la variante.
- IMPRECISE: indica que la posición de la variante no se conoce con precisión.
- CIPOS / CIEND: intervalo de confianza alrededor de las posiciones POS y END para variantes imprecisas.

3.3 VG

En esta sección hablaremos de la herramienta VG [6], para entender su importancia se expondrá un contexto, se explicará que es un grafo de variación y donde nace la necesidad de que surgiera la herramienta.

3.3.1 Contexto

Actualmente para los genomas pequeños, es posible estudiar la variación genética ensamblando genomas completos y luego compararlos mediante la comparación del genoma completo.

Para genomas grandes, como el humano, el ensamblaje del genoma *de novo* completo y preciso no es práctico debido a la complejidad y escala repetidas, por tanto, se utiliza información previa para la interpretación de nuevos datos de secuencia en su contexto genómico correcto.

En la actualidad las prácticas a seguir consisten en alinear las lecturas de secuencia con una única secuencia del genoma de referencia de alta calidad que presenta un haplotipo en cada ubicación del genoma. Aunque es mucho más rápido que ensamblando y simplificando el descubrimiento y la notificación de variantes genéticas. Este enfoque conduce a mapear sesgos hacia variantes que coinciden con la secuencia de referencia y se alejan de variantes alternativas, incluso habrá alguna secuencia en cada nueva muestra que está completamente ausente en la referencia, para evitar dichos sesgos, los datos deberían estar alineados con una secuencia de referencia "personalizada" que ya incorpora las variantes del individuo, pero en general no se sabe que variantes están presentes en una muestra antes de alinear los datos a partir de ella.

Sin embargo, la mayoría de las diferencias entre cualquier genoma y la referencia se segregan en la población, concluyendo que una estructura de referencia que representa una variación compartida conocida contendrá la mayor parte de la secuencia personalizada correcta para cualquier individuo.

La estructura computacional natural para hacer esto es el grafo de secuencia o estructuras equivalentes, se han utilizado anteriormente para representar múltiples secuencias que contienen diferencias o ambigüedades compartidas en una sola estructura, por ejemplo, múltiples alimentos de secuencia tienen una representación natural como grafos de secuencia parcialmente ordenados.

Es aquí donde entra en juego el formato VCF formato del que hablamos en el apartado anterior y podríamos entender como la definición de un grafo parcialmente ordenado similar a los implicados por una alineación de secuencia múltiple. Las estructuras relacionas que se utilizan con frecuencia en el ensamblaje del genoma incluyen el grafo De Brujin y el grafo de cadenas, que colapsan largas secuencias repetidas, por lo que los mismos nodos se utilizan para diferentes regiones del genoma.

3.3.2 ¿Qué es un grafo de variación y cómo se compone?

Un grafo de variación [16] combina tres tipos de elementos en una estructura de datos pangenómica. Tenemos secuencias de ADN (nodos), enlaces permitidos entre ellos (bordes) y genomas (caminos) que son recorridos por el grafo. Los nodos tienen identificadores, que son numéricos, y las rutas tienen nombres, que son cadenas de texto. Se hace una concesión para reflejar el uso genómico de estos grafos. Son bidireccionales y representan ambas cadenas de ADN, por lo que las posiciones se refieren a la orientación del complemento directo o inverso de los nodos. Esto significa que hay cuatro tipos de aristas (+ / +, +/-, - / -, - / +), cada una de las cuales implica su propio complemento inverso.

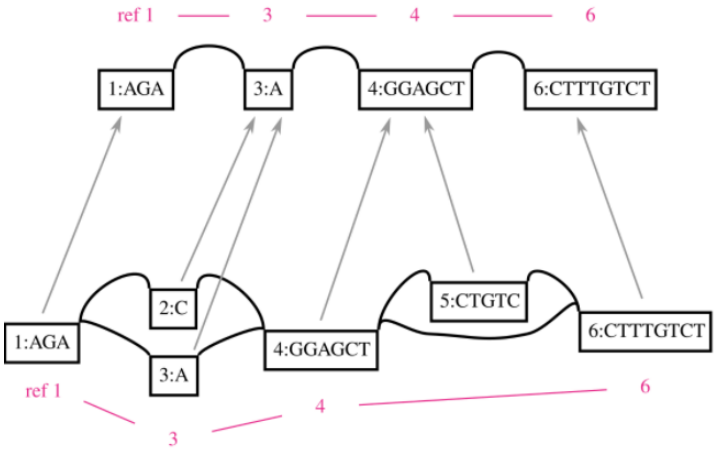


Figura 3.1: Salida de grafos de VG.

3.3.3 Características internas de la herramienta

La implementación de VG es multiproceso y está escrita en C++. Está disponible en el repositorio siguiente de github github.com/vgteam/vg en su versión 1.33 bajo licencia de software de código abierto del MIT. Proporciona una aplicación principal para respaldar las

operaciones que describimos aquí y una biblioteca libvg que las aplicaciones pueden usar para acceso a las estructuras de datos, índices y operaciones de bajo nivel.

La representación principal del grafo utiliza el sistema protobuf [17] de código abierto de Google, que admite directamente la serialización en el disco para su almacenamiento. También proporcionamos un formato de alineación protobuf, GAM, con una funcionalidad análoga a BAM, pero también podemos exportar asignación con respecto a las referencias incrustadas en formato BAM.

Para habilitar el mapeo de lectura y otras operaciones de acceso aleatorio contra grandes grafos de secuencia, han implementado una representación sucinta de un grafo de variación de VG (xg) que es estático pero muy eficiente en memoria y tiempo, para ello se han utilizado diccionario de clasificación/selección y otras estructuras de datos de la biblioteca SDSL. Los grafos se pueden importar y exportan en una variedad de formatos de intercambio gráficos.

Para las alineaciones, VG utiliza cualidades base en las puntuaciones de alineación y calcula las puntuaciones de calidad de mapeo ajustadas.

Capítulo 4 Herramientas de visualización

4.1 Sequence Tube Maps

Las estructuras de datos de grafos permiten la codificación de múltiples secuencias relacionadas en una única estructura de datos. La intención es simplificar la comparación de múltiples secuencias facilitando la búsqueda de similitudes y diferencias entre las secuencias. Hay varios enfoques (y formatos de archivo) para codificar formalmente variantes de secuencias genómicas y sus relaciones en forma de grafos. Desafortunadamente, a menudo es difícil visualizar estos grafos de una manera que transmita la información compleja pero que sea fácil de entender.

Por esta razón se lanza Secuence Tube Map [7], que es un módulo de JavaScript para la visualización de grafos de secuencia genómica. Genera automáticamente una visualización similar a un "mapa de tubo" de los grafos de secuencia que se han creado con VG. Esta aplicación está disponible online vgteam.github.io/sequenceTubeMap/ y se podrá instalar como se verá en el notebook.

4.1.1 Funcionalidad

El propósito de este módulo es generar representaciones visuales de grafos de secuencia genómica. La visualización tiene como objetivo mostrar la información sobre todas las variantes de secuencia de una manera intuitiva y lo más elegante posible.

Los grafos de secuencia genómica constan de nodos y rutas:

- Un **nodo** representa una secuencia específica de bases. La longitud de esta secuencia determina el ancho del nodo en la pantalla gráfica.
- Una **ruta** conecta varios nodos. Cada ruta representa una de las secuencias subyacentes a la estructura de datos del grafo y su recorrido por múltiples nodos.

Este sencillo ejemplo muestra dos caminos a lo largo de tres nodos:

Figura 4.1: Ejemplo sencillo Secuence Tube Map

Dado que ambas rutas conectan los mismos nodos, sus secuencias son idénticas (y los tres nodos podrían fusionarse en uno solo). Si las dos secuencias difirieran en algún punto intermedio, esto resultaría en la siguiente imagen:

Figura 4.2: Nodos con secuencias que difieren

La forma en que cambian las secuencias genómicas en los organismos vivos puede llevar a la inversión de subsecuencias. Para estos casos, en lugar de crear dos nodos diferentes, un solo nodo se atraviesa en dos direcciones diferentes:

Figura 4.3: Nodos con inversión de subsecuencias

El módulo Sequence Tube Map usa estos elementos como bloques de construcción y automáticamente establece y dibuja visualizaciones de grafos que son mucho más grandes y complicados.

4.2 MoMI-G

Es un navegador de grafos de genoma integrado modular de múltiples escalas. La secuenciación de lectura larga permite un descubrimiento más sensible y preciso de variantes estructurales, lo que requiere filtrar y validar miles de candidatos. Debido a que la mayoría de las herramientas de visualización muestran un único rango de una secuencia lineal, no son adecuadas para variantes estructurales grandes (sobre megabase) o anidados.

Además, ningún explorador de genoma existente permite a los usuarios inspeccionar simultáneamente las alineaciones de lectura que abarcan el alelo de referencia y un alelo alternativo causado por una variante estructural heterocigoto.

Por eso llega MoMI-G [8], un navegador gráfico de genoma para visualizar variantes estructurales en los grafos de variación, que proporciona una vista basada en gráficos que muestra un genoma con ramas y alineaciones en ellos. Los usuarios pueden filtrar, visualizar con anotaciones genómicas e inspeccionar SV con alineaciones de lectura. Se puede obtener desde el siguiente vínculo github.com/MoMI-G/MoMI-G

Figura 4.4: Logo MoMI-G

4.2.1 Funcionalidad

El propósito de este módulo al igual que el anterior es el de generar representaciones visuales de grafos de secuencia genómica. La visualización tiene como objetivo mostrar la información sobre todas las variantes de secuencia de una manera intuitiva y lo más elegante posible, para ello MoMI-G nos oferta tres tipos de visualización de grafos:

- Escala de cromosomas: Los arcos son cromosomas, las curvas representan las variaciones estructurales.

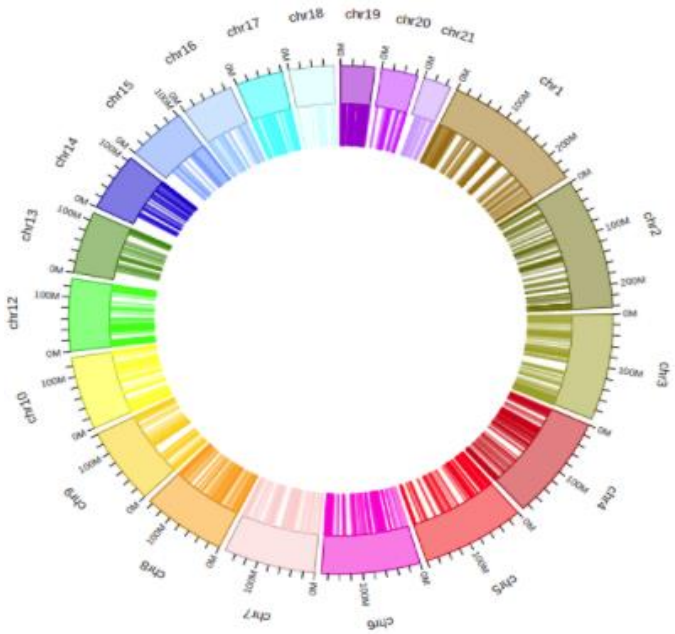


Figura 4.5: Grafico de escalas de cromosomas MoMI-G

- **Escala de genes:** Representa la misma información que en Sequence Tube Map; es más, utiliza Sequence Tube Map dentro de la aplicación.

- **Escala de nucleótidos:** La información se representa a través de nucleótidos A, G, C y T.

Figura 4.6: Grafico de escalas de nucleótidos MoMI-G

Capítulo 5 Desarrollo de los Jupyter Notebook

5.1 Resultados obtenidos de VG

Obtener los resultados de VG es algo más complicado que con el resto de los programas porque son ficheros con extensiones no tan comunes y los editores comunes no optimizan su lectura y los muestran de manera compleja. Por ello, simplemente haremos referencias a los ficheros generados en el cuaderno y, si fuera posible, añadiremos captura del contenido del fichero.

5.1.1 Resultados en diferentes formatos

- Construcción del grafo que genera un archivo llamado x.vg siendo su visualización ilegible para el usuario. El siguiente comando:

  vg construct -r x.fa -v x.vcf.gz > x.vg

  genera un archivo llamado x.vg que contiene el grafo de variación obtenido desde un archivo de origen x.vcf.gz.

- Convertidor a otros formatos, por ejemplo, en el cuaderno hemos usado la salida a pdf a través de Graphviz. Para ello, usamos primero el comando para formatearlo a .dot y a continuación usaremos la herramienta Graphviz para generar un pdf, también se pueden generar otros formatos como, JPG, PNG o incluso JSON con el grafo de variación a continuación se muestra imagen. Los comandos que generan dichas salidas son los siguientes:

  vg view -d x.vg > x.dot y dot -Tpdf x.dot > x.pdf

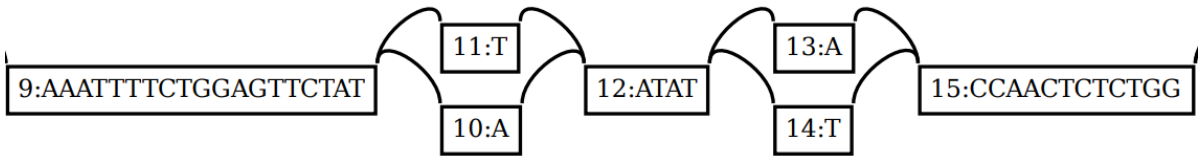


Figura 5.1: Gráfico que muestra el grafo de variación en formato de PDF

- El resto de los comandos que se hacen referencia en el cuaderno no genera fichero exceptuando la alineación. Por ello, no se ha hecho referencia a ellos ya que hacen modificaciones directamente sobre el archivo "x.vg" donde se almacena el grafo de variación.

5.2 Resultados de Sequence Tube Map

5.2.1 Resultados de diferentes trazas.

- Podemos observar parte de una traza con los nodos redundantes eliminados, con las lecturas

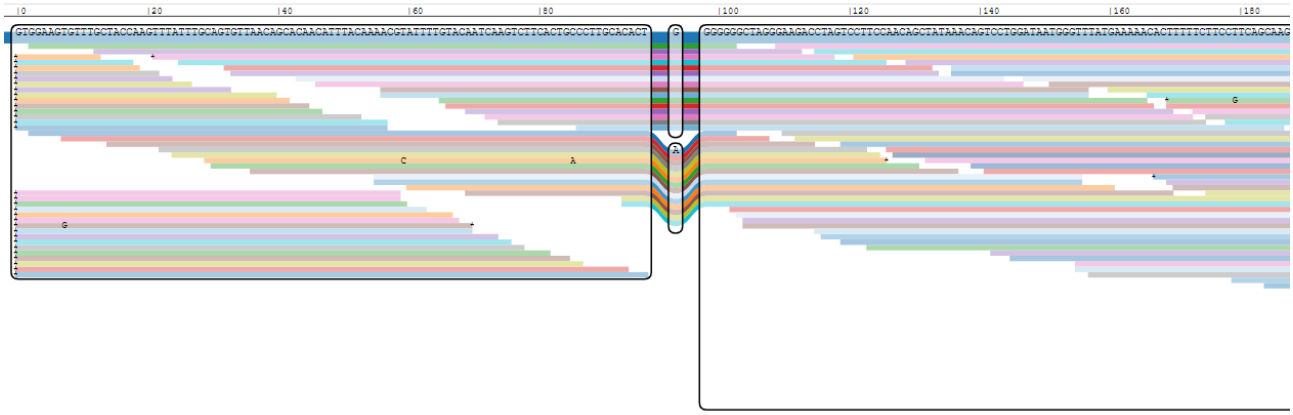


Figura 5.2: Gráfico color full para el data set snp1kg-BRCA1

- A continuación, veremos parte de una traza en escalas de grises, una traza sin eliminar los nodos redundantes y con la vista en versión comprimida.

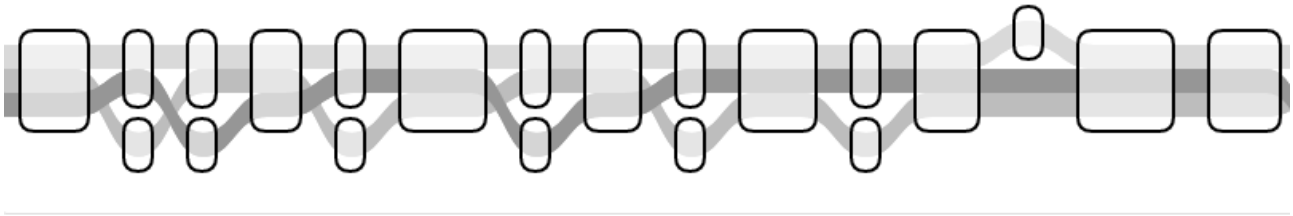


Figura 5.3: Gráfico en escala de grises para el data set small vg

- Representación de inserciones y polimorfismos solamente

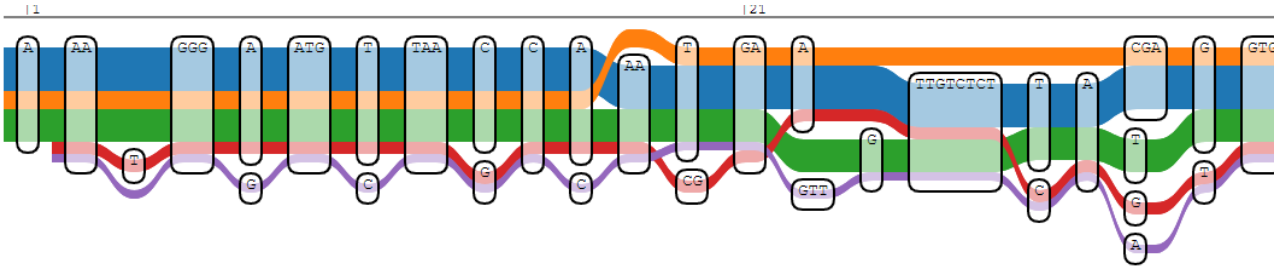


Figura 5.4: Gráfico que representa las inserciones y polimorfismos synthetic data

- Representación de las inversiones

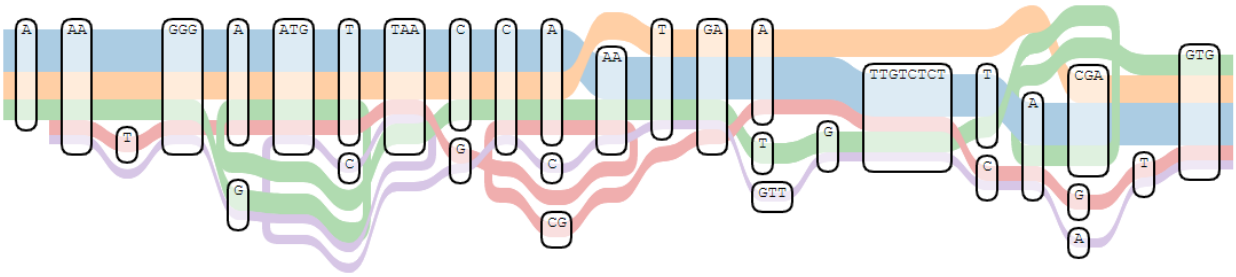


Figura 5.5: Gráfico que representa las inversiones pales colors synthetic data

- Representación de las inversiones anidadas

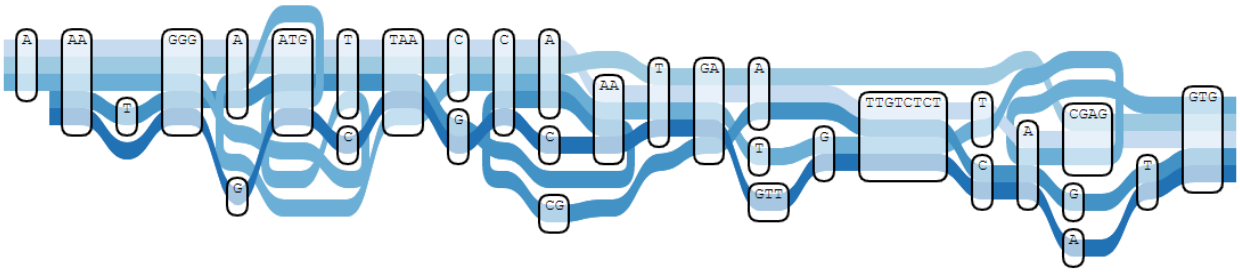


Figura 5.6: Gráfico que representa las inversiones anidadas blue colors synthetic data

- Representación de las duplicaciones

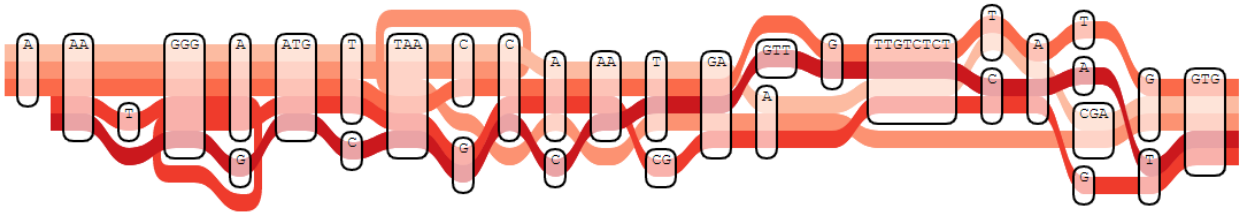


Figura 5.7: Gráfico que representa las duplicaciones red colors synthetic data.

- Representación de las translocaciones

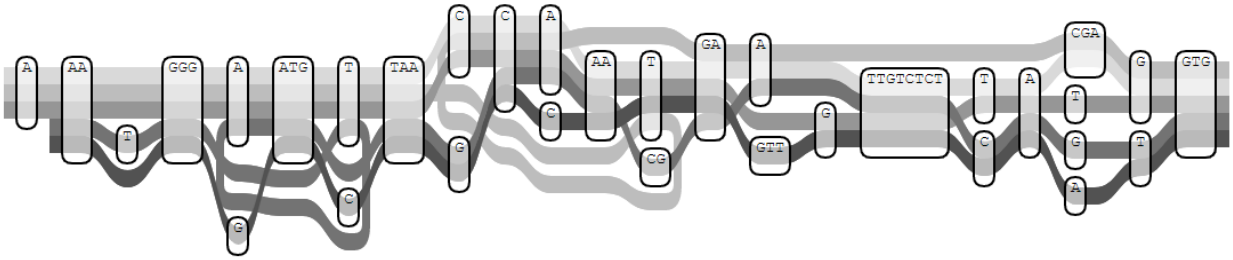


Figura 5.8: Gráfico que representa las translocaciones grey colors synthetic data

- Parte de la representación de las lecturas alienadas

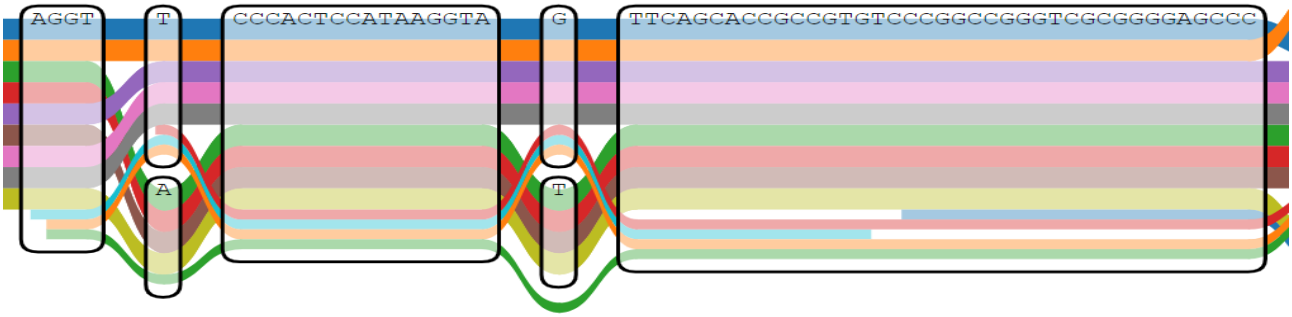


Figura 5.9: Gráfico que representa parte de las lecturas alineadas synthetic data.

5.3 Resultados de MoMI-G

5.3.1 Resultados de diferentes trazas

- Representación de todos circular de todos los cromosomas:

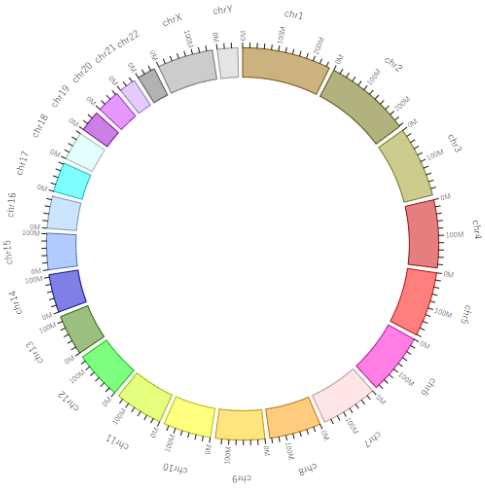


Figura 5.10: Gráfico que representa todos los cromosomas de la secuencia CHM1

- Representación circular de solos cromosomas pares:

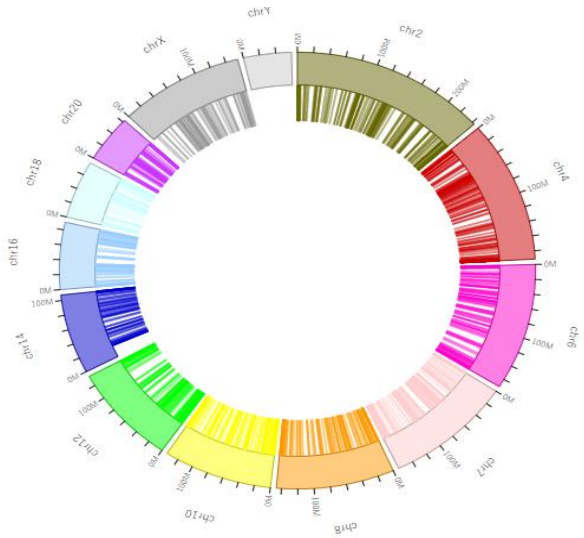


Figura 5.11: Gráfico que representa todos los cromosomas pares de la secuencia CHM1

- Representación circular con variaciones estructurales:

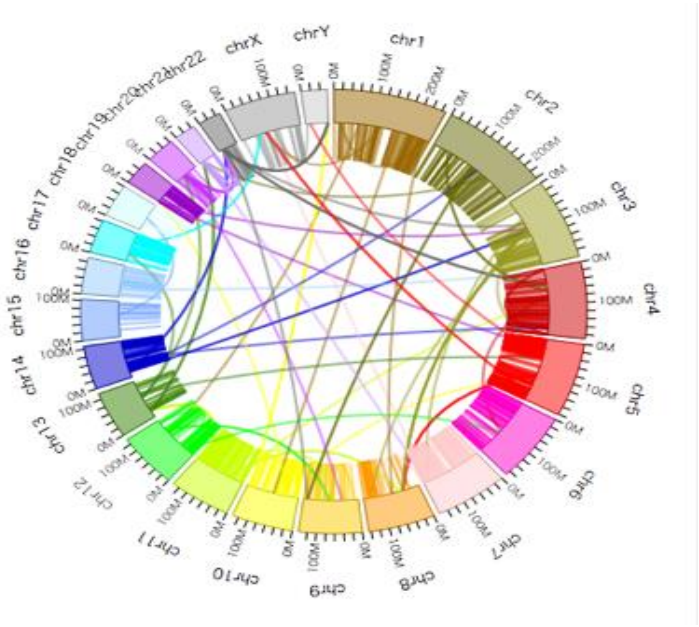


Figura 5.12: Gráfico circular con representación de variaciones estructurales de la secuencia CHM1

- Representación del cromosoma 12 a través de Sequence Tube Map integrado en MoMI-G

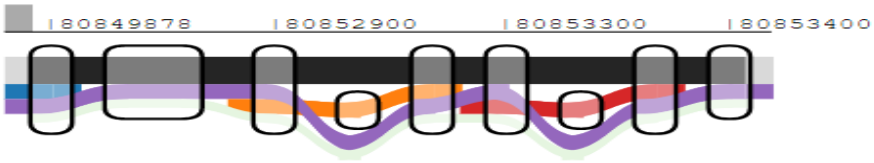


Figura 5.13: Gráfico que representa el cromosoma 12 a través de Sequence Tube Map de la secuencia CHM1

- Representación del cromosoma 12 con nucleótidos en Sequence Tube Map integrado en MoMI-G

Figura 5.14: Representación el cromosoma 12 con nucleótidos a través de Sequence Tube Map de la secuencia CHM1

5.4 Dockerización de las herramientas

La dockerización de las herramientas se ha optado por no realizarlo porque ya existe imagen oficial de Jupyter, en Docker Hub. Por tanto, hacer un contenedor con Jupyter sería replicar una imagen ya existente, y lo mismo ocurre con los visualizadores o la herramienta VG.

Por otro lado, también cabe a destacar que todos los cuadernos están preparados para funcionar en un entorno Linux, entorno que tiene por defecto los contenedores Docker, volviendo a estar en la tesitura de hacer algo que ya está desarrollado y no tiene sentido para el usuario complicarle el uso de las herramientas.

Para finalizar se concluye que no es necesario dockerizar las herramientas porque los cuadernos ya las descargan y las levantan de manera local sin problema en casi cualquier entorno de Linux.

Capítulo 6 Conclusiones y líneas futuras.

La realización de este Trabajo de Fin de Grado ha sido inmensamente gratificante por el hecho de los conocimientos adquirido en una rama como la bioinformática, rama en la que nunca me hubiese imaginado tener una vinculación. Así es, este Trabajo de Fin de Grado me ha portado aumentar más la capacidad de conocimiento, maneras nuevas de investigar, así como el incremento de la compresión de la información encontrada, análisis de esta y sobre todo enriquecerme como ingeniero.

Las tecnologías año a año se hacen más presentes en nuestras vidas, por tanto, la vinculación de la tecnología y la secuenciación genómica era obvio que también haya llegado al mismo punto. Es por ello por lo que se ha avanzado enormemente en cuanto a la implicación de la tecnología en este campo. Hace diez años los costes de secuenciación, así como la interpretación de la información era costosa y compleja. Gracias a la tecnología esto ha cambiado totalmente.

Todos estos avances se sintetizan en cada vez entender mejor la información del cuerpo humano por tanto esto implica hacer cosas tan increíbles como prevenir enfermades genéticas o mucho mejor comprender esas enfermedades raras para poder generar tratamientos en el que hoy en día no existen, entender mejor el cáncer y por supuesto la actual pandemia que está en el planeta.

A lo largo del todo el documento se han expuesto la realización de diferentes cuadernos Jupyter donde se explica al usuario como poder ejecutar las distintas herramientas para la generación y visualización de los grafos de variación, facilitando así el uso de dichas herramientas, solo debiendo tener instalado la plataforma Jupyter para poder desplegarlas y hacer un uso de ellas básico.

Llegar a ese punto no ha sido tarea fácil. Se ha tenido que investigar mucho, adquirir bastantes conocimientos nuevos sobre todo en la rama de la biología, mejorar mi compresión lectora en otro idioma, así como lidiar con algunos errores que expondré ahora.

El primer escollo que se encontró fue la constante evolución del núcleo de los cuadernos: la herramienta VG. Esta herramienta está al pie de cañón, hay muchos cambios de versión y a veces estos cambios afectan en las maneras en como genera la información, en los archivos de salida o en los de entrada, en la manera en la que se leen produciendo variaciones entre una versión y otra, y generando errores.

En un cambio de versión de la herramienta VG se actualizó la manera de alineación de la secuencia, ocasionando problemas en la herramienta de visualización Sequence Tube Map. Esto se descubrió a través de que se intentara levantar la herramienta de visualización y no se lograra de manera correcta, ya que no podía generar los grafos de tuberías porque la versión de VG se había actualizado, cambiando la manera de interpretar la información de

la alineación y produciendo un error en la generación del grafo de tuberías. Esto se reportó a los creadores de la aplicación y solucionaron la manera de leer los índices con la nueva versión de VG.

El segundo punto de inflexión fue dentro de los propios cuadernos y de cómo incluir en el PATH la herramienta VG. La herramienta es necesaria para poder levantar los dos visualizadores tanto MOMI-G como Sequence Tube Maps. Por tanto, es necesario que esta herramienta la localice en el PATH a la hora de levantar ambas aplicaciones. Esto se ha realizado en los tres cuadernos para poder tener una ejecución limpia y sin problemas en cualquier máquina de entorno UNIX, evitando así que el usuario tenga que realizar nada más allá que ejecutar el cuaderno.

Finalmente se encontró una última problemática a la hora de usar la bash dentro de los cuadernos, daba un error al ejecutar aplicaciones directamente haciendo que dichas aplicaciones no se ejecutan o se ejecutaran con errores hacia el usuario que podría causar confusión. Para ello se añadió el comando la opción de comando "--no-raise-error", evitando así la problemática.

En un futuro se podría plantear la integración de estas herramientas expuestas en el Trabajo de Fin de Grado con otras que también interpretan la secuenciación ya sea a través de formatos como el VCF o los grafos de variación en una sola herramienta para que los investigadores tenga a mano diferentes formatos para la interpretación de la información obtenida y facilitar así su trabajo, en un único paquete.

Capítulo 7 Summary and Conclusions.

The completion of this Final Degree Project has been immensely gratifying due to the knowledge acquired in a branch such as bioinformatics, a branch in which I would never have imagined having a link. Thus, this Final Degree Project has allowed me to increase my knowledge capacity, new ways of researching, as well as increasing the understanding of the information found, its analysis and, above all, enriching me as an engineer.

Technology is becoming more and more present in our lives every year, therefore, the link between technology and genomic sequencing has obviously reached the same point, which is why there has been enormous progress in terms of the involvement of technology in this field. Ten years ago, the costs of sequencing, as well as the interpretation of the information, were expensive and complex. Thanks to technology, this has changed completely.

All these advances are synthesized in an ever better understanding of the information in the human body, which means doing incredible things such as preventing genetic diseases or better understanding rare diseases in order to generate treatments that do not exist today, better understanding cancer and, of course, the current pandemic that is affecting the planet.

Throughout the whole document, different Jupyter notebooks have been presented, explaining to the user how to execute the different tools for the generation and visualisation of variation graphs, thus facilitating the use of these tools by only having to have the Jupyter platform installed to be able to deploy them and make basic use of them.

Getting to this point has not been an easy task, I have had to do a lot of research, acquire a lot of new knowledge especially in the field of biology, improve my reading comprehension in another language, as well as deal with some errors that I will expose now.

The first obstacle that was encountered was the constant evolution of the core of the notebooks, the VG tool, since this tool is at the bottom of the barrel and there are many version changes and sometimes these changes affect the ways in which it generates the information, in the output files or in the input files, and in the way they are read producing variations between one version and another, generating errors.

In a change of version of the VG tool, the way in which the sequence alignment was updated, causing problems in the Sequence Tube Map visualisation tool, this was discovered when an attempt was made to raise the visualisation tool and it was not able to raise it correctly, as it could not generate the pipe graphs because the version of VG had been updated, changing the way of interpreting the alignment information producing an error in the generation of the pipe graph. This was reported to the creators of the application and they solved the way of reading the indexes with the new version of VG.

The second turning point was within the notebooks themselves and how to include the VG tool in the PATH. The tool is necessary to be able to run both MOMI-G and Sequence Tube

Maps, therefore it is necessary to locate this tool in the PATH in order to run both applications. This has been done in the three notebooks in order to have a clean and problem-free execution in any UNIX environment machine, thus avoiding the user having to do anything more than run the notebook.

Finally a last problem was found when using the bash inside the notebooks, it gave an error when executing applications directly causing that these applications were not executed or were executed with errors to the user that could cause confusion, for it was added the command option "--no-raise-error", thus avoiding the problem.

In the future, we could consider combining these tools described in the Final Degree Project with others that also interpret sequencing through formats such as VCF or variation graphs in a single tool so that researchers have different formats at hand for the interpretation of the information obtained and thus facilitate their work, in a single package.

Capítulo 8 Presupuesto

Para obtener el presupuesto total, se plantea los siguientes gastos reflejados en las siguientes tablas.

8.1 Costes materiales

Material o Licencia	Precio
Licencias de software utilizado para el proyecto.	0 €
Licencia Microsoft Office 365	60 €
Servidor	2000 €
Ordenador portátil.	750 €

Tabla 8.1: Resumen presupuesto material y licencias

8.2 Costes de recursos humanos

Tarea	Horas	Precio	Total
Documentación y aprendizaje.	150	18€/h	2.700 €
Desarrollo de los cuadernos.	80	18€/h	1.440 €
Redacción de la memoria	32	18€/h	576 €

Tabla 8.2: Resumen presupuesto Ingeniero Informático

8.3 Costes totales

Recursos	Coste
Recursos materiales	2.810€
Recursos humanos	4.676€
Total	7.486€

Tabla 8.3: Resumen total del presupuesto

Bibliografía.

[1] Bioinformática

https://microbenotes.com/bioinformatics-introduction-and-applications

[2] Secuenciación de Sanger
https://bioinf.comav.upv.es/courses/intro_bioinf/sanger.html

[3] Secuenciación de nueva generación

https://es.khanacademy.org/science/high-school-biology/hs-molecular-genetics/hs-biotechnology/a/dna-sequencing

[4] Jupyter

https://jupyter.org/

[5] Docker

https://www.docker.com/

[6] VG

https://github.com/vgteam/vg

[7] Sequence Tube Map

https://github.com/vgteam/sequenceTubeMap

[8] MoMI-G

https://github.com/MoMI-G/MoMI-G

[9] VCF

http://samtools.github.io/hts-specs/VCFv4.3.pdf

[10] Graphviz

https://graphviz.org/

[11] FASTQ format

https://en.wikipedia.org/wiki/FASTQ_format

[12] BAM

https://informatics.fas.harvard.edu/whole-genome-resquencing-for-population-genomics-fastq-to-vcf.html

[13] Variation calling

https://informatics.fas.harvard.edu/whole-genome-resquencing-for-population-genomics-fastq-to-vcf.html

[14] VCFtools

http://vcftools.sourceforge.net/

[15] GAM

https://en.wikipedia.org/wiki/Genome_architecture_mapping#:~:text=In%20molecular%20biology%2C%20genome%20architecture,in%20a%20ligation%20independent%20manner.&text=GAM%20is%20the%20first%20genome,of%20genomic%20loci%20without%20ligation.

[16]Grafo de variación

https://ekg.github.io/

[14] Protobuf

https://en.wikipedia.org/wiki/Protocol_Buffers