UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería

COVLNS: Una *pipeline* bioinformática para la identificación de variantes de SARS-COV-2 en el Laboratorio Nacional de Salud

Trabajo de graduación presentado por Esteban Del Valle Campollo para optar al grado académico de Licenciado en Ingeniería Bioinformática

Guatemala,

2022

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería

COVLNS: Una *pipeline* bioinformática para la identificación de variantes de SARS-COV-2 en el Laboratorio Nacional de Salud

Trabajo de graduación presentado por Esteban Del Valle Campollo para optar al grado académico de Licenciado en Ingeniería Bioinformática

Guatemala,

2022

Vo.Bo.:

(f) _____

MSc. Augusto Franco

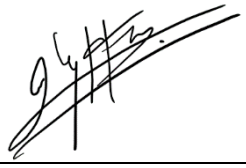
Tribunal Examinador:

(f) _____

MSc. Augusto Franco

(f) _____

Ing. Douglas Barrios



(f) _____

MSc. Jorge Chang

Fecha de aprobación: Guatemala, 7 de diciembre de 2022.

Prefacio

Este trabajo representa la culminación de cinco años de estudios universitarios que trajeron consigo una infinidad de experiencias, aprendizajes, obstáculos y momentos que marcarán el resto de mi vida. A pesar de que casi 2 años de este tiempo se vieron afectados por una pandemia, pude encontrar valor en el encierro y mi carrera universitaria no se atrasó ni deterioró de mayor manera. La pandemia trajo consigo aprendizajes para todo el mundo y dio a nacer oportunidades que no hubieran sido posibles sin ella. Este proyecto, así como la experiencia que viví en el Laboratorio Nacional de Salud mientras lo realizaba y la gente que conocí, no hubieran sido posibles si no fuera por la pandemia.

A pesar de su importante rol, la pandemia no dio lugar a este proyecto por si sola. Es importante para mí, agradecerles primeramente a mis padres, quienes me apoyaron y motivaron desde el día en que nací para seguir este camino en el que estoy infinitamente agradecido de estar. Mi familia y amigos que estuvieron conmigo durante todo el camino también forman parte importante de este trabajo y no posible sin su apoyo.

Le agradezco a la Universidad Del Valle de Guatemala y al Laboratorio Nacional de Salud por ser las instituciones que me permitieron llevar a cabo este trabajo y me dieron la oportunidad de aprender y desarrollarme haciendo lo que me apasiona. Agradezco a todos los profesionales de estas instituciones que me aconsejaron, apoyaron y guiaron durante la duración de este proyecto.

Para mí, es un honor, un privilegio y un placer poder presentar este proyecto de graduación y optar al grado de licenciatura en ingeniería bioinformática. Espero de corazón que este proyecto sea de utilidad para las personas para las que fue diseñado y que impulse a Guatemala más adelante, por más poco que sea, en el camino de la ciencia y la tecnología.

Por último, hago una dedicatorio especial de este proyecto a mi abuelo, Francisco Campollo, que falleció a sus 96 años durante la elaboración de este informe. El siempre creyó en mi capacidad de ser un profesional y tener éxito en el campo de mi elección. Que este trabajo sirva para iniciar a probar que estaba en lo correcto y cumplir con las expectativas que el tenía.

Índice

Lista de figuras

Lista de cuadros

Resumen

La secuenciación es una técnica utilizada en el proceso de obtener una secuencia de nucleótidos que conforman una cadena de ADN, ARN o proteína. El ADN (ácido desoxirribonucleico) es el material genético en las células que contiene la información utilizada en el desarrollo y funcionamiento de todos los organismos vivos conocidos. Se compone de 2 polímeros largos (hebras) que a su vez se componen de unidades llamados nucleótidos. Estos nucleótidos son guanina, adenina, citosina y timina. El ARN (ácido ribonucleico) es similar al ADN, pero es comúnmente de una sola hebra y contiene uracilo en lugar de timina (ThomasJeffersonUniversity, 2020). Como parte del proceso para obtener la secuencia de nucleótidos, es necesario analizar los datos obtenidos de la secuenciación mediante el uso de varias herramientas bioinformáticas. El Laboratorio Nacional de Salud (LNS) de Guatemala, recientemente inauguró el área de secuenciación con el objetivo de realizar vigilancia genómica de SARS-CoV-2. El objetivo de este proyecto es desarrollar e implementar una pipeline computacional para el análisis genómico de datos de secuenciación de muestras de SARS-COV-2. Esto ayudará al Laboratorio Nacional de Salud y apoyará a la población guatemalteca en el área de salud pública. Se estudiaron distintas pipelines utilizadas en análisis genómico alrededor del mundo para adaptarlas e implementarlas dentro del LNS. Se realizó un diseño previo y se llevaron a cabo pruebas de funcionalidad antes de iniciar la fase se desarrollo. En la fase de desarrollo se escribieron scripts en R que generan gráficas de cobertura para realizar control de calidad. También se escribieron programas en bash y se modificaron las pipelines existentes para unificarlas y adaptarlas a las necesidades del LNS. Por último, se realizaron pruebas con datos reales producidos por el personal del laboratorio para obtener resultados y verificar la funcionalidad. Se logró desarrollar un programa en Linux simple y funcional que presenta varios controles de calidad, así como resultados concisos de manera útil y comprensible. Se recomienda utilizar un software para crear pipelines y adaptar el programa para que funcione de manera universal.

CAPÍTULO 1

Introducción

La secuenciación es una técnica utilizada en el proceso de obtener una secuencia de nucleótidos que conforman una cadena de ADN o los aminoácidos que conforman una proteína. Esta técnica consiste en fragmentar una muestra de ADN en varios pedazos y añadir nucleótidos complementarios con florescencia de cierto color. Luego de esto, el equipo identifica los colores de la muestra fluorescente para poder identificar el orden de nucleótidos (KhanAcademy, 2016). No obstante, esta innovadora técnica por si sola, no es capaz de identificar de manera confiable la secuencia completa. El resultado al secuenciar una muestra, son archivos en formato fastq que no tienen orden. Para obtener un resultado conciso, es necesario someter estos resultados a una serie de pasos para lograr analizarlos, procesarlos, y convertirlos en resultados útiles e interpretables.

Eso es posible mediante el uso de la bioinformática y una diversa cantidad de herramientas computacionales asociadas a la misma. Luego de la secuenciación se realiza control de calidad y el ensamblaje de las lecturas. El ensamblaje genera contigs, que a su ves generan scaffolds, que a su vez pueden generar secuencias de consenso. Sin embargo, realizar este proceso puede ser un proceso largo, tedioso y complicado, especialmente para personas sin experiencia en uso de software en Linux y línea de comando. La cantidad de programas y tiempos de espera necesarios también puede hacer que este proceso sea muy largo (Gladman, s.f.). A partir de las secuencias de consenso generadas, se pueden llevar a cabo aún más análisis bioinformáticos que permiten obtener más información genómica. Por ejemplo, análisis para identificar variantes de diferentes virus y especies de seres vivos.

El Laboratorio Nacional de Salud (LNS) de Guatemala, recientemente inauguró el área de secuenciación con el objetivo de realizar vigilancia genómica de SARS-CoV-2 y conocer qué variantes están circulando a lo largo del país. Para facilitar el análisis de datos y recopilación de resultados, se plantea desarrollar una pipeline o flujo de trabajo e implementarla en el LNS. De esta manera, el personal podrá obtener resultados confiables y comprensibles luego del proceso de secuenciación.

CAPÍTULO 2

Justificación

El análisis de datos de secuenciación puede ser un proceso largo y tedioso. Un secuenciador solo devuelve una serie de datos incomprensibles y desordenado. Para poder obtener resultados útiles, es necesario procesarlos de cierta manera utilizando una variedad de herramientas de bioinformática. Estas herramientas, normalmente basadas en Linux con uso en la terminal, pueden tener una documentación limitada y/o difícil de entender. Adicionalmente, normalmente presentan una gran cantidad de problemas al ser instaladas y construir el ambiente. Esto se amplifica para las personas que no tienen experiencia utilizando este tipo de software o la terminal de una máquina de Linux.

Existen múltiples pipelines creadas que tienen el objetivo de facilitar todo el proceso de análisis bioinformático. Estas pipelines proveen una alternativa más simple a utilizar muchas herramientas y analizar el gran volumen de datos generado para seguir el progreso. No obstante, estas pipelines pueden ser complicadas de usar para muchas personas ya que siguen teniendo una gran cantidad de dependencias y diversas complejidades. Además, todas las pipelines varían en cuanto a su funcionamiento y los resultados finales que devuelven. Estas pipeline también tienden a devolver una gran cantidad de documentos como resultados que, dependiendo del objetivo final del usuario, pueden no usarse y solo ocupan espacio.

En el Laboratorio Nacional de Salud, se tiene la importante labor de llevar un control epidemiológico de las variantes de SARS-COV-2 que están circulando en todo el país. Para esto, es necesario secuenciar muestras de todo el país y analizar los datos utilizando bioinformática. El proceso de análisis puede ser complejo y largo, disminuyendo la eficiencia con la que se presenta esta importante información al público. Por esto, es de gran importancia poder facilitar el proceso con el que se analizan estos datos y se obtienen los resultados deseados.

Para esto, se considera que la mejor solución es la implementación de una pipeline personalizada al formato de trabajo del Laboratorio Nacional de Salud. Esta pipeline recibe de manera directa los datos del secuenciador utilizado, los procesa automáticamente y devuelve de manera ordenada y comprensible solamente los resultados que son de utilidad para el personal del Laboratorio y de interés para el usuario. De esta manera, se facilita el proceso,

se vuelve más eficiente, y se ocupa menos espacio en los equipos del laboratorio.

Actualmente, el Laboratorio analiza los datos mediante la plataforma web del proveedor de servicios, pudiendo provocar sesgos. Además, se desconoce el método por el cual estos son analizados por ser un programa cerrado. Otra limitante al utilizar plataformas web es la conexión inestable a internet, pudiendo tener problemas en la carga de datos.

CAPÍTULO 3

Objetivos

3.1. Objetivo general

Desarrollar una pipeline para el análisis genómico de datos de secuenciación de muestras de SARS-COV-2 con el objetivo de ayudar al Laboratorio Nacional de Salud y apoyar a la población guatemalteca en el área de salud pública.

3.2. Objetivos específicos

- Hacer una pipeline sencilla que cumpla con los estándares de calidad.
- Automatizar el análisis para hacer todo el proceso más eficiente.
- Presentar resultados de manera compresible, gráfica y útil para el Laboratorio y el personal de secuenciación.

CAPÍTULO 4

Marco teórico

4.1. Virus SARS-CoV-2

El virus SARS-CoV-2 es un virus miembro de una gran familia de virus llamados coronavirus. Estos virus pueden infectar a personas y algunos animales. Se supo por primera vez que el SARS-CoV-2 infectaba a las personas en 2019. Se cree que el virus se propaga de persona a persona a través de las gotas que se liberan cuando una persona infectada tose, estornuda o habla. También se puede propagar al tocar una superficie que tiene el virus y luego tocarse la boca, la nariz o los ojos, pero esto es menos común. El SARS-CoV-2 también es el virus responsable de causar la enfermedad respiratoria COVID-19. Esta enfermedad puede causar síntomas como fiebre, dificultad para respirar, toz, dolor de cuerpo y puede conllevar a la muerte del paciente (NIH, s.f.).

El genoma de los CoV (27-32 kb) es un ARN monocatenario de sentido positivo (+ssRNA) que es más grande que cualquier otro virus de ARN. La proteína de la nucleocápside (N) forma la cápside fuera del genoma y el genoma se empaqueta aún más mediante una envoltura que está asociada con tres proteínas estructurales: proteína de membrana (M), proteína de espiga (S) y proteína de envoltura (E). Como miembro de la familia de los coronavirus, el tamaño del genoma del SARS-CoV-2 que se secuenció recientemente es de aproximadamente 29,9 kb. El SARS-CoV-2 contiene cuatro proteínas estructurales (S, E, M y N) y dieciséis proteínas no estructurales (nsp116) (Wang et al., 2020).

4.1.1. Pandemia

En diciembre del 2019, el virus SARS-CoV-2 fue identificado por primera vez en la provincia de Wuhan, China. La enfermedad COVID-19, rápidamente se esparció a lo largo de toda China y el mundo (Zhu et al., 2020). Los altos números de contagios y la tasa de mortalidad ocasionada por el virus ocasionaron que esta enfermedad fuera declarada como pandemia global el 11 de marzo del 2020 por la Organización Mundial de la Salud (Cucinotta

y Vanelli, 2020).

Mediante progresaba la pandemia, muchos países empezaban a observar bajas en la cantidad de casos, hospitalizaciones y muertes debido a las restricciones que los ciudadanos habían sufrido para evitar la propagación del virus. Sin embargo, hacia fines del verano, en agosto de 2020, la variante Lambda se descubrió por primera vez en Perú. Hasta la fecha, esta variante se ha extendido a al menos 29 países, según la OMS.

Un mes después, la variante Alpha se identificó por primera vez en el Reino Unido en septiembre de 2020. El descubrimiento de estas variantes fue significativo, mostró que el virus estaba evolucionando. Como resultado, los síntomas y consecuencias de la enfermedad estaban cambiando. La evidencia ha demostrado, por ejemplo, que la variante Alpha puede presentar un mayor riesgo de síntomas graves de COVID-19.

Con la aparición de estas nuevas variantes, los casos de COVID-19 comenzaron a aumentar nuevamente en muchos países y el 29 de septiembre de 2020, había 1 millón de muertes por COVID-19 (Moore, 2021).

En Guatemala, hasta la fecha de revision de este informe (diciembre 2022), han habido 1,177,088 casos acumulados de COVID-19 con 19,969 fallecidos registrados. Esto significa una incidencia de 6,982.2 por 100,000 habitantes, una tasa de mortalidad de 118.5 por 100,000 habitantes y una letalidad de 1.7 %. El 53 % de los casos corresponde a personas de sexo femenino y el resto a personas de sexo masculino. La capital muestra ser el lugar con una mayor incidencia seguida por los departamentos vecinos de sacatepeqeuz y el progreso (MSPAS, 2022).

4.1.2. Variantes y diferenciación

Todos los virus, incluido el SARS-CoV-2, cambian con el tiempo debido a mutaciones en su genoma. Esto se da ya que a medida que un virus se replica, sus genes sufren .^{er}rores de copia.^{a}leatorios. Con el tiempo, estos errores de copia genética pueden, entre otros cambios en el virus, provocar alteraciones en las proteínas o antígenos de la superficie del virus (CDC, 2022). La mayoría de los cambios tienen poco o ningún impacto en las propiedades del virus. Sin embargo, algunos cambios pueden tener efectos significativos sobre el virus, como la facilidad con la que se propaga, la gravedad de la enfermedad asociada o el rendimiento de las vacunas, los medicamentos terapéuticos, las herramientas de diagnóstico u otras medidas sociales y de salud pública. A finales del año 2020, la aparición de variantes que planteaban un mayor riesgo para la salud pública mundial impulsó la caracterización de variantes de interés y variantes de preocupación (VOI y VOC por sus siglas en inglés) específicas. Esto, con el fin de priorizar el seguimiento y la investigación a nivel mundial e informar la respuesta en curso a la Pandemia de COVID-19 (OMS, 2021).

Actualmente, existen 5 VOCs, de las cuales una se encuentra en circulación. Las variantes Alpha, Beta, Gamma y Delta se consideran variantes previamente en circulación y la variante Ómicron se considera como en circulación a la fecha de publicación de este documento (OMS, 2021). La variante alpha es asociada con mayor transmisibilidad. La variante beta muestra indicios de se mas dificil para ser destruida por anticuerpos. La variante gamma es similara la Beta. La variante Delta muestra ser mas facil de contagiar. La variante Omicron muestra

un mayor riesgo de reinfección (Shah, 2021). En la Figura 1, se puede observar un diagrama con las diferentes VOCs, su fecha de identificación, su linaje asociado y una descripción de sus efectos.

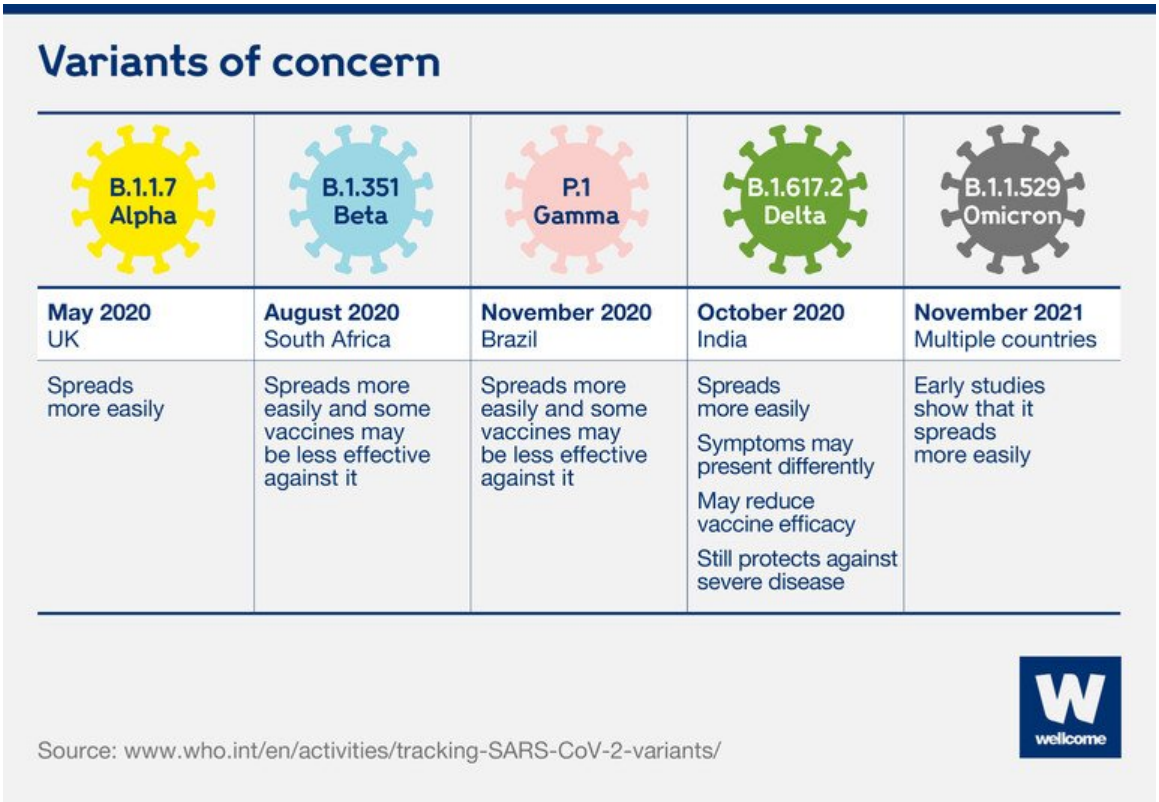


Figura 1: Diagrama de variantes de SARS-CoV-2

(Shah, 2021)

La existencia de estas variantes generó la necesidad de identificar la variante del virus con la que un paciente puede estar infectado y así poder llevar un control genómico de la enfermedad. La secuenciación del genoma completo, o al menos la secuenciación completa o parcial del gen de pico es el mejor método para caracterizar una variante específica (ecdc, 2022).

4.2. Secuenciación genómica

Secuenciación es un proceso usado para descifrar el material genético que se encuentra en un organismo o virus. Este proceso se utiliza para eventualmente obtener la secuencia ordenada de nucleótidos que componen una cadena de ADN o ARN. Las secuencias de las muestras se pueden comparar para ayudar a los científicos a rastrear la propagación de un virus, cómo está cambiando y cómo esos cambios pueden afectar la salud pública (CDC, 2020). Existen varios tipos de secuenciación que se han desarrollado a lo largo de los años, en el Laboratorio Nacional de Salud se utiliza secuenciación de nueva generación o NGS por

sus siglas en inglés.

4.2.1. Proceso

Primero, se debe de preparar la muestra. Para esto, se fragmenta la secuencia de en pedazos de tamaño similar y se agregan adaptadores. Luego, se carga la muestra en una flow cell. Una flow cell es una placa con millones de pocillos capaces de capturar un solo fragmento de ADN utilizando los adaptadores. Una vez este la muestra en la flow cell, se ingresa al secuenciador. Aquí, cada fragmento se copia mediante PCR para crear un cluster y obtener sitios más densos. Luego, la enzima polimerasa agrega una base complementaria, teñida de un color especifico dependiendo de la base y se toma una fotografía de la flow cell. Después, se lava el colorante y agrega una segunda base complementaria al cluster. Este ciclo se repite entre 100 y 200 veces para obtener la secuencia completa (Zhang, 2022). A continuacion, se inlcuye un diagrama en donde se puede observar como funciona el proceso de secuenciacion de manera general.

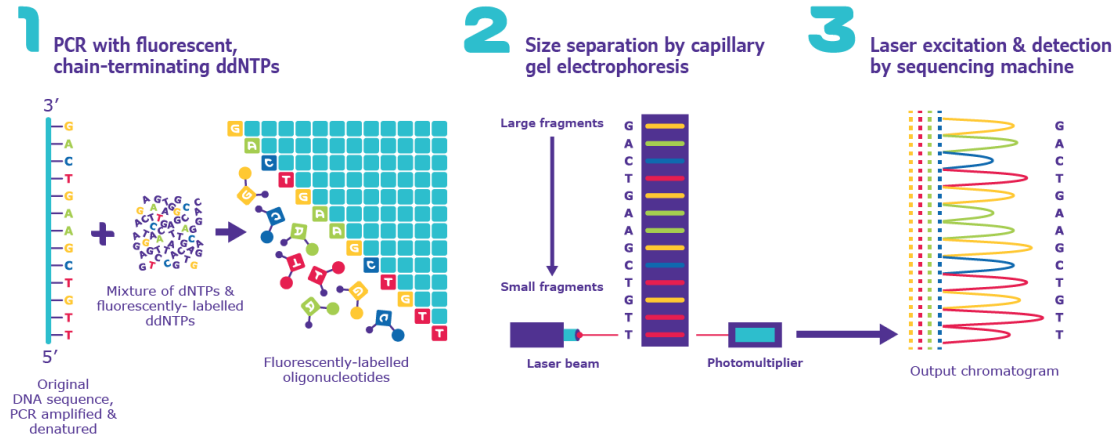


Figura 2: Diagrama de funcionamiento de secuenciación

(MERCK, 2022)

Una vez termina este proceso, se obtiene una enrome cantidad de datos, ya que se secuencian millones de fragmentos a la vez. Para procesar estos datos y obtener un producto final, se debe de realizar un análisis computacional utilizando bioinformática (Zhang, 2022). Este proceso puede no ser exacto en ocasiones y pueden existir fallos en varias etapas del proceso, desde la preparación de muestras hasta dentro del secuenciador, que ocasionen que se obtengan resultados erróneos y difíciles de procesar. Uno de los mayores desafíos que acompañan a la tecnología NGS es el mayor riesgo de descubrir variantes de significado clínico desconocido. La gran cantidad de genes que se analizan puede conducir a una serie de hallazgos no deseados, como factores de riesgo para otras enfermedades o variantes no clasificadas. Esto también puede ser considerado como una ventaja de esta tecnología (Fahrioğlu, 2018).

4.3. Análisis bioinformático

Los archivos generados por el secuenciador (normalmente en formato fastq) no tienen utilidad práctica por sí solos. Es necesario procesar estos datos y transformarlos de manera que se puedan generar resultados útiles y comprensibles acorde a lo que se está buscando. En este caso, se busca hacer un ensamblaje de genoma y un análisis de variantes (Kalyanaraman et al., 2011).

4.3.1. Ensamblaje

Primero, se realiza el ensamblaje. El ensamblaje del genoma es el proceso computacional de descifrar la composición de la secuencia del material genético (ADN) dentro de un organismo, utilizando numerosas secuencias cortas llamadas lecturas derivadas de diferentes porciones del ADN objetivo como entrada. Estas lecturas se obtienen a partir de secuenciación (Kalyanaraman et al., 2011).

Primero, se debe de hacer un control de calidad de las lecturas. Esto sirve para comprender los datos crudos, tomar decisiones informadas sobre cómo manejarlos y maximizar las posibilidades de obtener un ensamblaje de buena calidad. El conocimiento de los tipos de lectura, la cantidad de lecturas, su contenido de GC, la posible contaminación y otros problemas son importantes. Esta información ayuda a identificar cualquier problema de calidad con los datos y en la elección de los métodos de limpieza de datos a utilizar. La limpieza de los datos crudos antes del ensamblaje puede conducir a ensamblajes de mejor calidad (Gladman, s.f.).

Una de las herramientas más populares para control de calidad de archivos fastq es FastQC. Esta herramienta funciona en diversos sistemas operativos y opera con una interfaz gráfica o mediante línea de comando. Esta herramienta devuelve varios parámetros de calidad útiles para el usuario. Algunos de los parámetros más importantes son:

- Longitud de lectura: será importante para establecer el valor de tamaño máximo de k-mer para el ensamblaje
- Tipo de codificación de calidad: importante para el software de recorte de calidad
- % GC: los organismos de GC altos no tienden a ensamblarse bien y pueden tener una distribución de cobertura de lectura desigual.
- Número total de lecturas: le da una idea de la cobertura.
- Caídas en la calidad cerca del comienzo, la mitad o el final de las lecturas: determina posibles métodos y parámetros de recorte/limpieza y puede indicar problemas técnicos con el proceso de secuenciación/ejecución de la máquina.
- Presencia de k-meros muy recurrentes: puede indicar la contaminación de las lecturas con códigos de barras, secuencias de adaptadores, etc.
- Presencia de un gran número de N en las lecturas: puede indicar un experimento de secuenciación de mala calidad. Debe recortar estas lecturas para eliminar las N.

(Andrews, 2010)

Una vez se tiene conocimiento sobre la calidad de la data cruda, se puede usar esta información para limpiar y recortar las lecturas para mejorar su calidad general. Esto significa que se pueden eliminar ciertas lecturas y cortar ciertos pedazos de las lecturas que afectan su calidad de manera negativa. Una de las herramientas más populares para limpieza es trimmomatic. Esta herramienta se utiliza en línea de comando y contiene varias funciones útiles que se pueden usar de manera secuencial para limpiar y recortar datos (Gladman, s.f.).

Con los datos limpios, se puede realizar el ensamblaje. Existen varios softwares que realizan este proceso. Por ejemplo, Velvet Optimiser, Spades, MIRA, ALLPATHS y SOAPdenovo. La mayoría de estos softwares funciona de manera similar. Su meta es producir largas piezas contiguas de secuencia (contigs) a partir de estas lecturas. A veces, los contigs se ordenan y orientan entre sí para formar scaffolds. A su vez, los scaffolds se pueden unir para producir una secuencia unificada. Es posible generar esta secuencia y/o verificar su calidad realizando una alineación con una secuencia de referencia (Gladman, s.f.).

4.3.2. Identificación de variantes

Con el genoma aensamblado, es posible realizar varios análisis útiles. Para este proyecto, el análisis de interés es la identificación y nomenclatura de diferentes linajes y variantes del virus SARS-CoV-2. Para este propósito, existe la herramienta Pango. Pango es un sistema basado en reglas para nombrar linajes genéticos de SARS-CoV-2. Proporciona una terminología común compartida para todos los que están investigando o discutiendo la transmisión y propagación del virus. Pango es un sistema dinámico y flexible que puede adaptarse a la naturaleza cambiante de la pandemia y al crecimiento en la generación de datos genómicos del SARS-CoV-2. La nomenclatura Pango es utilizada por investigadores y agencias de salud pública en todo el mundo y usa nombres de linaje Pango, como B.1.1.7 o B.1.351 (Rambaut et al., 2020).

Cada linaje Pango define un grupo de secuencias del genoma del SARS-CoV-2 y se crea de acuerdo con dos principios. Primero, los linajes Pango representan grupos o grupos de infecciones con ascendencia compartida. Si se puede pensar en toda la pandemia como un vasto árbol ramificado de transmisión, entonces los linajes Pango representan las ramas individuales dentro de ese árbol. En segundo lugar, los linajes Pango pretenden resaltar eventos epidemiológicamente relevantes, como la aparición del virus en una nueva ubicación, un rápido aumento de casos o la evolución de virus con nuevos fenotipos (Rambaut et al., 2020).

La nomenclatura Pango es un sistema jerárquico y eso se refleja en la forma en que se nombran los linajes. Cada linaje recibe un código alfanumérico único que contiene información parcial, pero no completa, sobre la historia filogenética de ese linaje. Las convenciones de nomenclatura de linaje representan un compromiso entre los requisitos de comprensión humana y la legibilidad de las máquinas.

Dentro de este sistema existen dos herramientas principales que son de interés. La primera es Pangolin. Pangolin fue desarrollado para implementar el sistema de nomenclatura pango.

Es una herramienta que permite al usuario asignar un genoma de SARS-CoV-2 su linaje más probable. La segunda herramienta es Scorpio. Esta es una herramienta para llamadas de variantes de preocupación basado en SNPs. Scorpio es útil para identificar las variantes de preocupación a partir de los linajes generados por Pangolin y devuelve el nombre común de la variante para que el usuario pueda entender el resultado (Rambaut et al., 2020).

4.4. Pipelines bioinformáticas

En informática, una pipeline es un conjunto de elementos de procesamiento de datos conectados en serie, de modo que la salida de un elemento es la entrada del siguiente. Sirven mediante una seria de pasos cronológicos que dan instrucciones específicas y automatizan un proceso más largo (DataPipelines, 2021). Las pipelines bioinformáticas son un componente integral de la secuenciación de próxima generación (NGS). El procesamiento de datos de secuencia sin procesar para detectar alteraciones genómicas tiene un impacto significativo en el manejo de enfermedades y la atención al paciente. Estas pipelines se componen de una amplia gama de algoritmos de software para procesar datos de secuenciación crudos y generar una lista de variantes de secuencia anotadas. Las pipelines bioinformáticas pueden ser diseñadas y desarrolladas por un proveedor con o sin personalización por parte del laboratorio o completamente desarrolladas por el laboratorio.

Una pipeline básica del exoma que ofrece variantes llamadas a partir de datos de secuenciación podría constar de tan solo 12 pasos, la mayoría de los cuales se pueden ejecutar en paralelo, pero un análisis real normalmente implicará varios pasos posteriores adicionales y la generación de informes complejos (Leipzig, 2017). A continuación, se muestra un diagrama de flujo que representa una pipeline para análisis de trío para detectar mutaciones de novo.

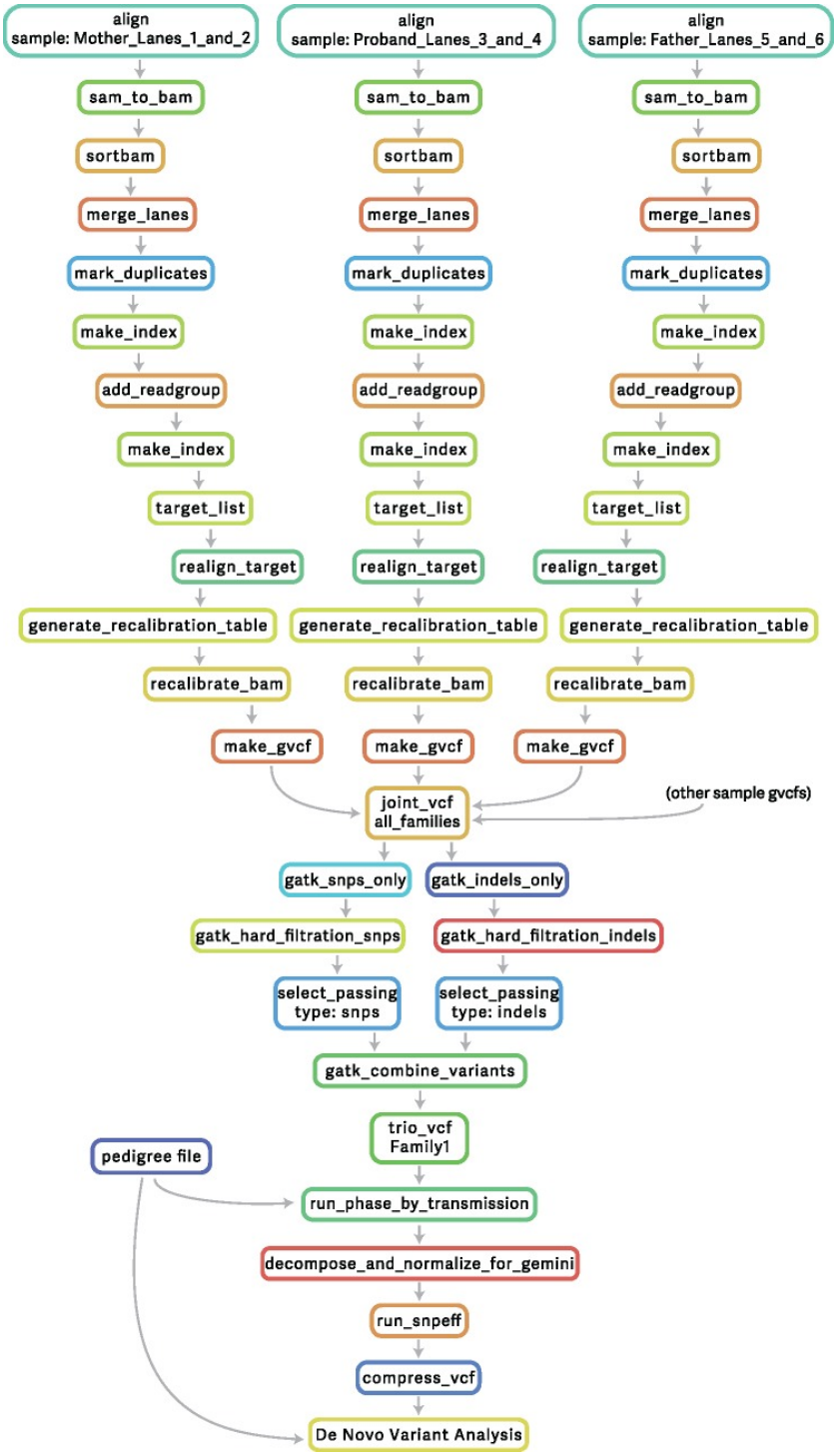


Figura 3: Diagrama de flujo que representa una pipeline para análisis de trío para detectar mutaciones de novo

(Leipzig, 2017)

Es posible observar que hay una gran cantidad de pasos para este proceso. Adicionalmente, se busca llevar a cabo análisis de tres diferentes muestras. Si no se utilizara una pipeline,

se debería de llevar a cabo cada paso y procesar cada muestra de manera individual. Esto Tomaría una gran cantidad de tiempo en comparación. Aunque las pipelines específicas de la bioinformática ofrecen análisis automatizados de alto rendimiento, no son frameworks en el sentido de que no son fácilmente extensibles para integrar nuevas herramientas definidas por el usuario (Leipzig, 2017).

4.4.1. Pipelines usadas en otros paises

Existen ya pipelines desarrolladas para la identifiacion de varaintes de SARS-CoV-2 utilizadas en diferentes partes del mundo. En este caso hay 3 pipelines que son de interes para este proyecto. HAvoC es una pipeline utilizada en Helsinki desarrollada por la universidad de Helsinki 12 (Truong Nguyen et al., 2021). Gencom, es utilizada por el Gorgas en Panama 14 (AAMCgenomics, 2022) y ViralFlow es utilizada por el Fiocruz en Brasil 13 (Dezordi et al., 2022). Estas pipelines funcionan de manera generalmente similar, como se muestra en el siguiente diagrama.

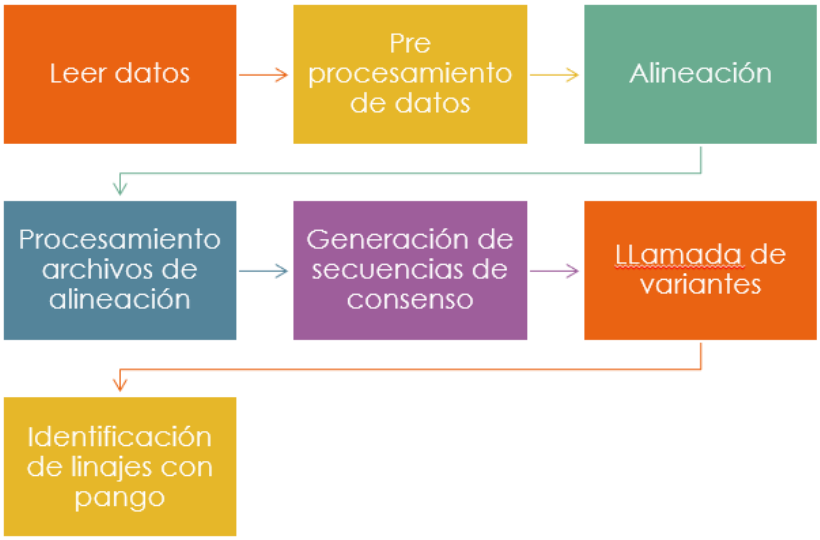


Figura 4: Diagrama de flujo que representa el funcionamiento general de las pipelines para identifiación de variantes de SARS-CoV-2

A pesar de sus similitudes, las pipelines mencionadas tienen varias diferencias en cuanto a su funcionamiento y su ejecución. Por ejemplo, HAVoC usa Fastp para control de calidad, hace llamada de variantes con la herramienta Lofreq, y usa ivar para eliminar primers y hacer llamadas de consenso. Trae sus propios archivos de referencia y su propio ambiente que hay que instalar previo a la ejecución. Luego, para correrlo, solo se llama y se le pasan los archivos fastq como parametro (Truong Nguyen et al., 2021). ViralFlow, de igual manera, usa fastp para control de calidad pero usa ivar para hacer la llamada de variantes. ViralFlow corre en un contenedor de Docker y se le pasa un archivo con extensión .conf al momento de correrlo en donde se especifican varios parámetros. Entre estos, los archivos de referencia que se utilizaran y la ubicación de los archivos fastq (Dezordi et al., 2022). Por último, Gencom no realiza ningun preprocesamiento de datos antes de alinear. Realiza anotaciones y un

control de calidad posterior a alinear con quasitools hydra y hace la llamada de variantes con MicroGMT. Esta es una particularidad de Gencom ya que MicroGMT no es una librería instalable. Es un repositorio en GitHub y hay que descargar todo el código fuente y modificar las instrucciones dentro del código fuente de Gencom para utilizarlo. Gencom también genera sus propias graficas de cobertura (AAMCgenomics, 2022). Los diagramas de flujo completos para cada pipeline estan inlcuidos en la sección de anexos de este informe 12.1.

4.5. Control genómico de COVID19 en Guatemala

En Guatemala, la generación de datos de secuenciación genómica se lleva a cabo mediante la Red Regional de Vigilancia Genómica de COVID-19, a través del Laboratorio Nacional de Salud (LNS) del Ministerio de Salud Pública y asistencia Social (MSPAS). Este proceso dio inicio en el 2020 secuenciando muestras en el extranjero y desde entonces ha realizado las siguientes acciones:

- Vigilancia epidemiológica: Se realiza vigilancia epidemiológica para llevar a cabo acciones de detección oportuna y búsqueda de casos en municipios silenciosos o con incrementos de casos sustanciales. De esta manera se pueden garantizar los insumos para el diagnóstico e implementar estrategias de rastreo de contactos y seguimiento de casos ambulatorios.
- Comunicación de riesgo: Se busca difundir desde el nivel nacional hasta el comunitario el riesgo del incremento de casos de acuerdo con la identificación de nuevas variantes e implementar medidas de restricción a nivel nacional o local.
- Vigilancia genómica: Se envían muestras al LNS de personas que consulten los servicios de salud para realizar pruebas diagnósticas y que tengan historial de viaje reciente. Se priorizan envíos de muestras de pacientes con ciertas características tales como departamentos fronterizos o con alto turismo, supercontagios, reinfecciones, sintomatología no reportada y aumento de casos graves en niños. Si el área no cumple con estas características, el laboratorio recibe 10 % de las muestras para realizar vigilancia.

(Conde, 2021)

CAPÍTULO 5

Metodología

5.1. Origen de los datos

Para obtener las muestras que serán secuenciadas, el LNS slecciona muestras positivas de hisopado nasofaringeo para COVD-19 tomadas a lo largo de todo el país. Los diferentes centros de salud y hospitales envian pruebas positivas para que sean procesadas en el laboratorio junto con una boleta incluye datos como información del pacietne, sintomas y area de salud. Estas muestras se someten a un PCR para verificar el resultado positivo y se les asigna un código de identifiación para almacenarlo dentro de su base de datos. A partir de estas muestras, se elige un número específico de muestras para secuenciar tomando en cuenta factores como la disponibilidad de reactivos y que haya diversidad de areas, sexo y edad en las muestras (de Salud, 2022).

5.2. Diseño previo

Inicialmente, se tuvo una discusión con personal del laboratorio y personal de la Organización Panamericana de la Salud (OPS) para establecer las necesidades del Laboratorio y las funcionalidades con las que debía de cumplir el producto final. Se acordó que el programa iba a incorporar diferentes pipelines de análisis de variantes de SARS-CoV-2 que operaran de maneras distintas. Este programa debía de poder tomar los datos directamente del secuenciador y procesarlos de la manera más simple posible proporcionando diferentes análisis y filtros de calidad y, como resultado final, un archivo comprensible con el resultado de linaje y la variante encontrada por muestra.

Una vez se tenía establecidas metas concretas para el producto final, se realizó un análisis de las pipelines propuestas para incorporar en el producto final. La OPS recomendó Viral-

flow (Dezordi et al., 2022), una pipeline desarrollada y usada en el Fiocruz en Brasil. Por otro lado, el personal del laboratorio tenía un alto interés en la pipeline Gencom (AAMCgenomics, 2022), la cual es usada por el instituto Gorgas en Panamá. Por último, se exploró la pipeline HAVoC desarrollada por BMC Bioinformatics en Helsinki (Truong Nguyen et al., 2021). Se experimentaron las 3 pipelines con datos modelo para evaluar su uso y como presentaban los resultados. A partir de esto, se pudo visualizar que se podía agregar, cambiar y eliminar de cada pipeline y como adaptarlas al uso para el Laboratorio. Cabe mencionar que para lograr probar las 3 pipelines se necesitó hacer varios cambios a su código fuente en cuanto a bibliotecas utilizadas, instrucciones que involucraban directorios y recortes de código innecesario.

Para consolidar un diseño y un algoritmo de funcionamiento para el producto final, se realizaron varios diseños en papel en forma de diagramas, diagramas de flujo y algoritmos. De esta manera se pudo visualizar de una manera comprensible el orden de los procesos y otros aspectos a tomar en cuenta tales como el orden de los archivos, las diferentes direcciones y otros retos o aspectos de la programación a tomar en cuenta antes de iniciar el desarrollo.

Para este proyecto, se utilizó Linux como sistema operativo dada su facilidad para instalar las herramientas bioinformáticas necesarias. Se preparó un ambiente utilizando conda («Anaconda Software Distribution», 2020) en donde se instalaron todas las herramientas, bibliotecas y dependencias necesarias. En este ambiente, tanto las pipelines a incorporar como programa diseñado, podrían funcionar adecuadamente. Estas se encuentran listadas en el repositorio del proyecto en la sección de requerimientos.

Por último, antes de iniciar con el desarrollo del programa principal, era importante establecer un orden para el manejo de archivos ingresados y generados. Los varios diseños en papel ayudaron a planear como se iban a administrar estos archivos para luego solo crear las direcciones adecuadas en la máquina.

5.3. Desarrollo

Como parte de los requerimientos del personal del laboratorio, para el proceso del control de calidad, se requería la generación de graficas de cobertura. Previo al desarrollo del programa principal, se realizó un programa en R capaz de generar una gráfica de cobertura a partir de un archivo de profundidad generado por samtools usando la opción depth. Samtools recibe un archivo bam y genera un archivo con extensión tsv que luego se pasa al programa diseñado en R para generar una gráfica. Una vez se había probado este programa a pequeña escala con una muestra, se prosiguió al desarrollo principal del programa y su estructura donde se implementarían las gráficas.

Posterior a esto, se inició la creación del programa principal en bash script. Primero, se creó un mensaje y se implementaron tres posibles banderas correspondientes a las 3 pipelines que se implementarían. En esta parte también se implementó programación defensiva para que el comando funcionara con diferentes orden de banderas, mayúsculas y minúsculas y un modo por si no se cumplían con las condiciones iniciales.

Luego, se implementó el primer control de calidad del programa. Se utilizó fastqc (An-

drews, 2010) para realizar análisis de calidad previo al análisis y multiqc (Ewels et al., 2016) para desplegar el control de todos los datos de manera simultánea. Una vez realizado el control de calidad, se despliega el archivo multiqc y se le da la oportunidad al usuario de eliminar secuencias que no cumplen con la calidad que se busca.

Si se inicializa el programa con la bandera "h", el programa corre en el modo HAVoC. Este es el modo más simple en cuanto a implementación. Solo basta con correr el programa ingresando como parámetro el directorio donde se encuentran los archivos fastq.

Si se inicializa el programa con la bandera "v", el programa corre en el modo ViralFlow. Este modo requiere ciertas acciones antes de que pueda correr apropiadamente. Primero, se copian los archivos de referencia a el directorio donde se encuentran los archivos fastq. Luego, se crea un archivo con extensión .conf. En este archivo se incluyen instrucciones para la ejecución del programa ya que ViralFlow es una herramienta que se corre dentro de Docker. Este archivo facilita la configuración y ejecución del programa usando Docker. Por último, se construye el contenedor de Docker y se corre el programa dentro de Docker pasando como parámetro el archivo generado previamente. En adición, se corre un comando chmod para que el usuario tenga acceso a los resultados ya que ViralFlow los genera con permisos restringidos.

Si se inicializa el programa con la bandera "g", el programa corre en el modo Gencom. En cuanto a implementación dentro del programa principal, basta con correr el Gencom ingresando el directorio como parámetro. No obstante, se realizó una gran cantidad de modificaciones tanto al script principal de Gencom como a scripts modulares. Especialmente para adaptar directorios utilizados, estructuras de almacenamiento de datos y formato de los datos utilizados. El funcionamiento interno de estas tres pipelines no forma parte de la metodología de este proyecto, pero se puede consultar sus algoritmos en los anexos y en el github del proyecto.

Una vez termina el proceso de una de las tres pipelines, se procede a hacer un último control de calidad utilizando graficas de cobertura. Gencom genera sus propias graficas por lo que no es necesario generarlas. Para los modos ViralFlow y HAVoC, se utilizó el script desarrollado previamente para generar estas gráficas. Cabe mencionar que el eje x se transforma a una escala logarítmica para obtener una mejor visualización. Una vez se tienen las gráficas, se despliegan y se le da al usuario una vez la oportunidad de revisar los datos y borrar los que no desea que se incluyan en los resultados finales.

La última parte del proceso es desplegar los resultados obtenidos de manera comprenisble y unificada. De manera similar a las gráficas, este proceso solo es necesario para ViralFlow y HAVoC ya que Gencom ya genera un CSV general para todas las muestras. Se toman los resultados CSV individuales de cada muestra y se unifican en un solo archivos CSV donde se presentan todas las muestras junto a el linaje obtenido y la variante identificada. Esto se realiza mediante un script simple que navega los directorios de resultados y copia la información al CSV general. Esta información se despliega y el programa finaliza.

En esta fase del proyecto, también se tomó la decision de que el programa funcionaría solamente en la máquina en donde fue diseñada. Esto se realizó de esta manera para darle un enfoque mayor a la consistencia de funcionamiento de todos los modos. El manejo de directorios y diferentes versiones de librerías que se debe de tener para que el programa

funcione (un problema comun en bioinformática), puede ser complejo y fácil de alterar de manera que el programa deje de funcionar. Po esta razón, se tomo la decisión de darle un enfoque a que el programa funcionara de manera consistente dentro de la máquina del laboratorio sobre hacer que el prorgama fuera facilmente operable en otros equipos.

5.4. Post desarrollo e implementación

Al tener el programa desarrollado, se realizaron pruebas con datos reales para evaluar su funcionalidad. Al realizar estas pruebas, se encontraron varias fallas relacionadas a directorios, librerías, ambientes de conda y formato de datos. Por ejemplo, algunos arhivos no se guardaban donde se debia. otros archivos no se generaban por un error de librerias. En ocasiones, el programa fallaba y se generaban archivos incorrectos por un mal control de versiones en los ambientes utilizados. Del mismo modo, ciertos archivos con informacion importante se generaban erroneamente ocasionando que no se pudieran leer por etapas posteriores que los utilizan. Esto lideró a una fase de debugging y troubleshooting en la que se corrían pruebas en los diversos modos usando datos reales y se solucionaba cualquier error que se encontrara en el camino. Una vez el programa funcionaba de manera adecuada y persistente, era necesario presentar el programa al personal del laboratorio para asegurarse que cumplía con sus expectativas y capacitarlos para utilizarlo.

Se tuvo una reunión con el equipo de secuenciación y personal del área de virología del LNS en donde se demostró la funcionalidad del programa, se presentaron los algoritmos y se les indicó como utilizarlo. Por último, se realizó una corrida de prueba por parte del personal para solucionar dudas y para que pudieran utilizar el programa antes de entregárselos.

5.5. Evaluación de uso

Se dejó el programa finalizado en el laboratorio para que el personal lo utilizara y lo probara por un periodo extendido de tiempo. De esta manera el personal podría verificar la funcionalidad y utilidad del programa en donde sea necesario. Se mantuvo contacto con el personal del laboratorio para solicitar retroalimentación, fotografías o cualquier métrica que pudiera servir para evaluar el uso del producto final.

CAPÍTULO 6

Resultados

6.1. Datos

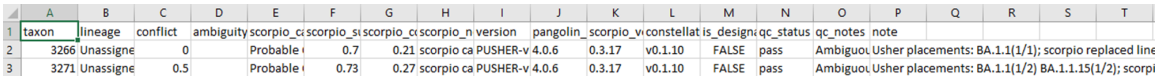
Se logró evaluar múltiples muestras de una secuenciación y presentar los resultados relativamente consistentes en un archivo csv de manera ordenada y entendible. Existen ciertas columnas que son de mayor relevancia. En la columna A, se observa el código asigando a la muestra. En la columna B se observa el linaje asignado. Las columnas C y D contienen coeficientes de confiabilidad del resultado. La columna E contiene la designación de la OMS de la variante encontrada. En la sección de anexos, se pueden encontrar tablas que muestran los datos usados como ejemplo con mejor resolución 12.6.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	taxon	lineage	conflict	ambiguity	scorpio_c	scorpio_s	scorpio_c	scorpio_n	version	pangolin_	scorpio_v	constellat	is_designa	qc_status	qc_notes	note	
2	3266	BA.1	0.57	0.89	Probable (	0.66	0	scorpio ca	PLEARN-v	4.0.6	0.3.17	v0.1.10	FALSE	pass	Ambiguous_content:0.07		
3	3271	BA.1	0.61	0.92	Probable (	0.68	0	scorpio ca	PLEARN-v	4.0.6	0.3.17	v0.1.10	FALSE	pass	Ambiguous_content:0.06		

Figura 5: Ejemplo de resultados finales del análisis en modo HAVoC de 2 muestras presentados en un archivo csv

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S
1	taxon	lineage	conflict	ambiguity	scorpio_c	scorpio_s	scorpio_c	version	pangolin_	pangoLEA	pango_ve	status	note						
2	3266_S75_	BA.1.1	0	0.963491	Omicron (	0.7119	0	PLEARN-v	3.1.20	2/2/2022	v1.2.124	passed_q	scorpio call: Alt alleles 42; Ref alleles 0; Amb alleles 14; Oth alleles 3						
3	3266_S75_	BA.1.1	0	0.963491	Omicron (	0.661	0.0339	PLEARN-v	3.1.20	2/2/2022	v1.2.124	passed_q	scorpio call: Alt alleles 39; Ref alleles 2; Amb alleles 13; Oth alleles 5						
4	3271_S76_	BA.1.1	0	0.993919	Omicron (	0.9322	0	PLEARN-v	3.1.20	2/2/2022	v1.2.124	passed_q	scorpio call: Alt alleles 55; Ref alleles 0; Amb alleles 1; Oth alleles 3						
5	3271_S76_	BA.1.1	0	0.993919	Omicron (	0.8644	0.0508	PLEARN-v	3.1.20	2/2/2022	v1.2.124	passed_q	scorpio call: Alt alleles 51; Ref alleles 3; Amb alleles 1; Oth alleles 4						

Figura 6: Ejemplo de resultados finales del análisis en modo ViralFlow de 2 muestras presentados en un archivo csv



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	taxon	lineage	conflict	ambiguity	scorpio_c	scorpio_s	scorpio_c	scorpio_n	version	pangolin_	scorpio_v	constellat	is_design	qc_status	qc_notes	note				
2	3266	Unassigne	0		Probable	0.7	0.21	scorpio ca	PUSHER-v	4.0.6	0.3.17	v0.1.10	FALSE	pass	Ambiguou	Usher placements: BA.1.1(1/1); scorpio replaced line				
3	3271	Unassigne	0.5		Probable	0.73	0.27	scorpio ca	PUSHER-v	4.0.6	0.3.17	v0.1.10	FALSE	pass	Ambiguou	Usher placements: BA.1.1(1/2) BA.1.1.15(1/2); scorpi				

Figura 7: Ejemplo de resultados finales del análisis en modo Gencom de 2 muestras presentados en un archivo csv

Se realizó un programa unificando e implementando diversas pipelines de manera ordenada y que no almacena archivos de manera que se genere desorden dentro de la máquina utilizada

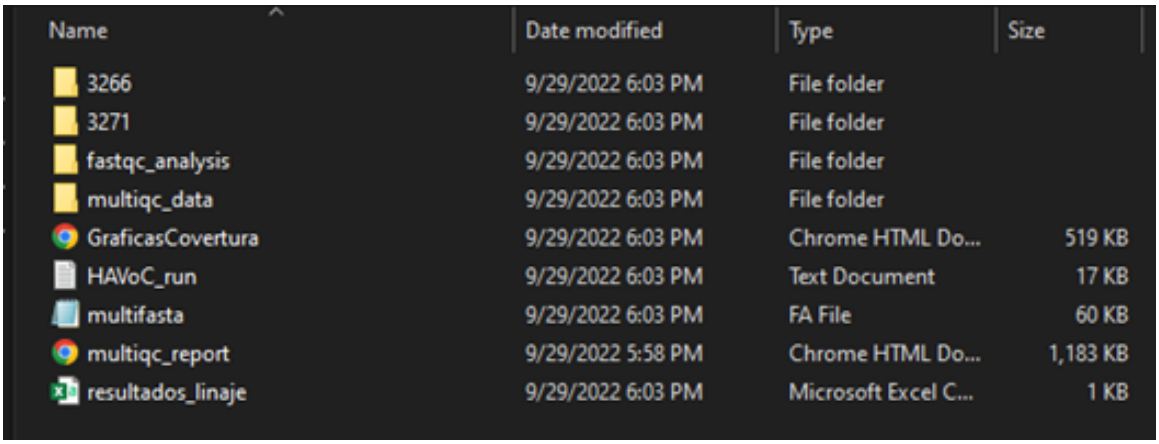
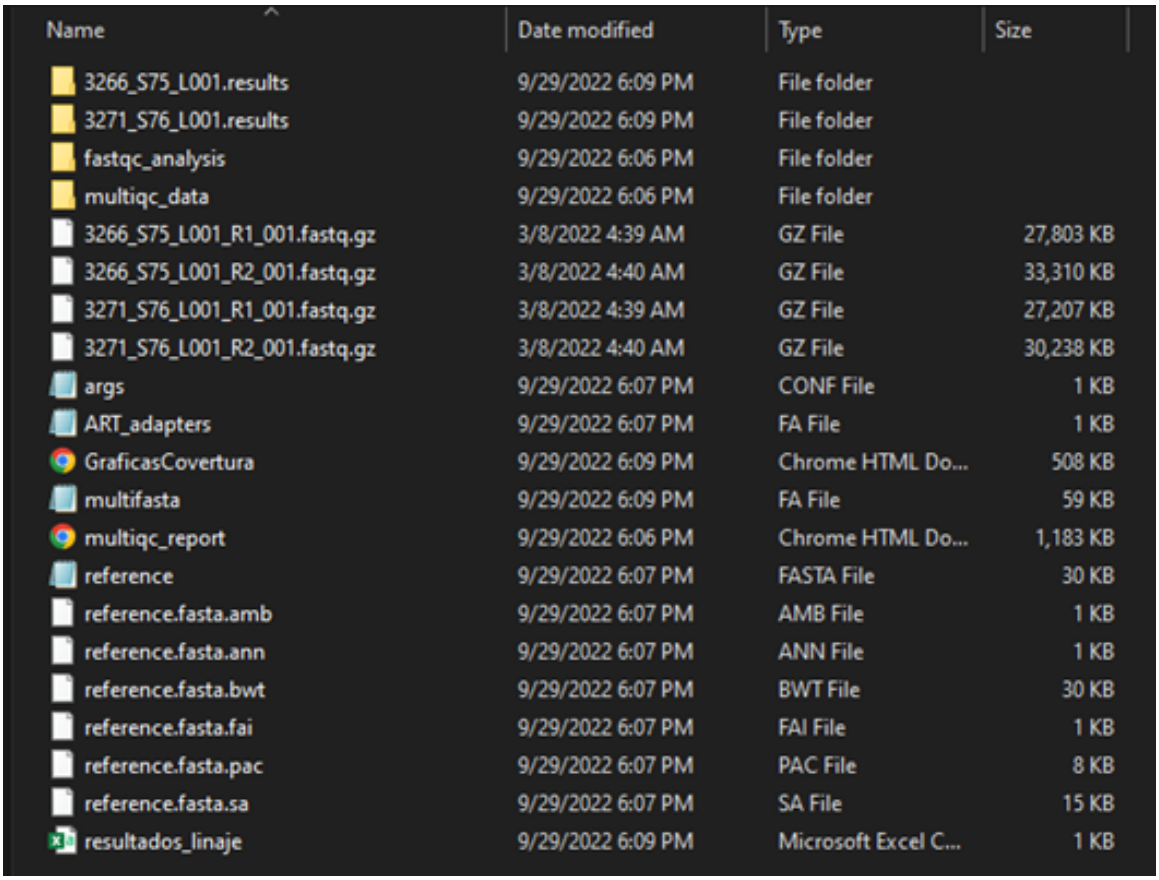
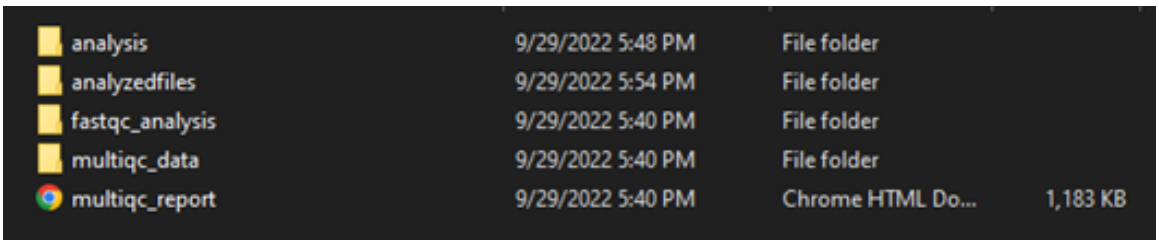


Figura 8: Almacenaje de datos finales en modo HAVoC



Name	Date modified	Type	Size
3266_S75_L001.results	9/29/2022 6:09 PM	File folder	
3271_S76_L001.results	9/29/2022 6:09 PM	File folder	
fastqc_analysis	9/29/2022 6:06 PM	File folder	
multiqc_data	9/29/2022 6:06 PM	File folder	
3266_S75_L001_R1_001.fastq.gz	3/8/2022 4:39 AM	GZ File	27,803 KB
3266_S75_L001_R2_001.fastq.gz	3/8/2022 4:40 AM	GZ File	33,310 KB
3271_S76_L001_R1_001.fastq.gz	3/8/2022 4:39 AM	GZ File	27,207 KB
3271_S76_L001_R2_001.fastq.gz	3/8/2022 4:40 AM	GZ File	30,238 KB
args	9/29/2022 6:07 PM	CONF File	1 KB
ART_adapters	9/29/2022 6:07 PM	FA File	1 KB
GraficasCovertura	9/29/2022 6:09 PM	Chrome HTML Do...	508 KB
multifasta	9/29/2022 6:09 PM	FA File	59 KB
multiqc_report	9/29/2022 6:06 PM	Chrome HTML Do...	1,183 KB
reference	9/29/2022 6:07 PM	FASTA File	30 KB
reference.fasta.amb	9/29/2022 6:07 PM	AMB File	1 KB
reference.fasta.ann	9/29/2022 6:07 PM	ANN File	1 KB
reference.fasta.bwt	9/29/2022 6:07 PM	BWT File	30 KB
reference.fasta.fai	9/29/2022 6:07 PM	FAI File	1 KB
reference.fasta.pac	9/29/2022 6:07 PM	PAC File	8 KB
reference.fasta.sa	9/29/2022 6:07 PM	SA File	15 KB
resultados_linaje	9/29/2022 6:09 PM	Microsoft Excel C...	1 KB

Figura 9: Almacenaje de datos finales en modo ViralFlow



analysis	9/29/2022 5:48 PM	File folder	
analyzedfiles	9/29/2022 5:54 PM	File folder	
fastqc_analysis	9/29/2022 5:40 PM	File folder	
multiqc_data	9/29/2022 5:40 PM	File folder	
multiqc_report	9/29/2022 5:40 PM	Chrome HTML Do...	1,183 KB

Figura 10: Almacenaje de datos finales en modo Gencom

Se lograron generar gráficas de cobertura que representan los datos de manera que se pueda verificar la calidad de los mismos.

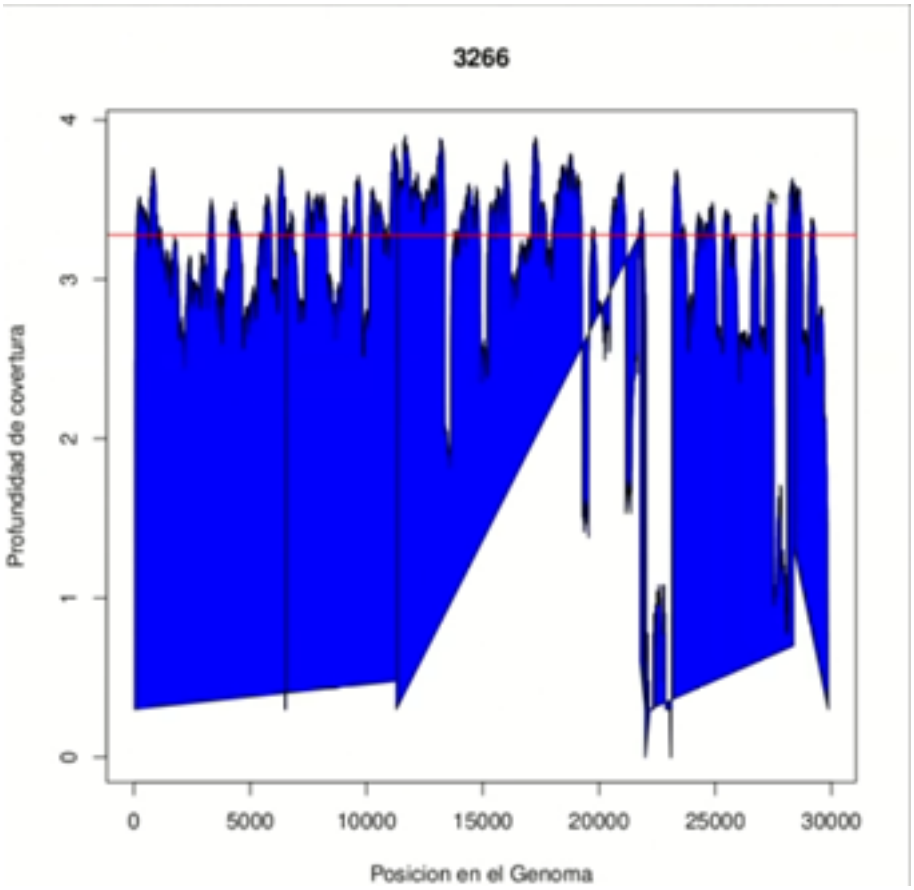


Figura 11: Gráfica de cobertura de una sola muestra. La gráfica se presenta en una escala logarítmica y la línea roja representa la media de los datos. El eje x representa cada posición en el genoma y el eje y muestra la profundidad de cobertura para cada posición.

6.2. Uso y aplicación

Se capacitó al personal del laboratorio para utilizar el programa y se observó que lo podían utilizar exitosamente para analizar datos.

Se logró que el programa funcionara de manera autónoma, requiriendo mínima participación de parte del usuario utilizando solamente un comando en la terminal.

El programa desarrollado tiene un impacto positivamente en el trabajo del laboratorio al generar resultados de confianza y utilidad sin dependencia de conexión de red. Los resultados de secuenciación y análisis genómico producidos por el LNS pueden ser observados en los informes que se publican. Estos informes pueden ser observados en el siguiente link:

http://portal.lns.gob.gt/index.php/component/sppagebuilder/77-inicio-2/

CAPÍTULO 7

Análisis de resultados

Como se puede observar en las figuras 1, 2 y 3, Los resultados de linaje de las muestras de secuenciación, se presentan en un CSV comprensible y útil para el personal del laboratorio en donde cada fila representa una muestra y cada columna un dato diferente. Entre todos los datos generados las 4 columnas con mayor importancia son: el nombre de la muestra, el linaje, la confiabilidad y la llamada de scorpio.

El nombre o id de la muestra es el código asignado a cada muestra por el laboratorio para llevar un control. El linaje es asignado por pangolín y representa la variante genómica de la muestra. La confiabilidad es con que certeza se le asigna cierto linaje a una muestra. Por último, la llamada de scorpio es el nombre común utilizado por la OMS que se la asigna aun linaje. Estos 4 datos son los de mayor utilidad para el personal del laboratorio ya que son los datos utilizados para generar reportes y poder llevar un control genómico apropiado de este virus en el país.

Se observa que, para cada uno de los modos, los resultados difieren en cuanto a linaje, pero existe cierto consenso en cuanto al resultado final. Para Gencom, no se pudo determinar un linaje de las muestras. Para ViralFlow y HAVoC se terminó el mismo linaje correspondiente a Ómicron. Los tres modos concluyen, aunque sea con cierta duda, que las muestran son de la variante ómicron.

En las figuras 4, 5 y 6, se puede observar cómo son almacenados los datos para cada uno de los modos. Se puede observar que difieren levemente en como se almacenan. Para los modos HAVoC y ViralFlow, se almacenan los resultados para cuada muestra de manera individual y el resultado final de linaje, así como las gráficas se encuentran en el directorio principal. Para ViralFlow, también se pueden observar los archivos originales, así como los archivos de referencia en el directorio principal. Para Gencom, se observan directorios diferentes para los resultados y los archivos originales. Dentro del directorio de resultados se agrupan los resultados en diferentes categorías en vez de por muestra.

En la Figura 7, se observa una gráfica de cobertura que se genera para el modo Viral-Flow y Gencom. Esta gráfica sufre una transformación logarítmica en el eje x para mejor visualización y posee una línea roja que representa la media de los datos. Esta gráfica forma parte de un archivo pdf generado con todas las gráficas de las muestras en la corrida y se inlcuyte en los resultados para ejemplificar como el usuario observa una de las métricas de calidad de la corrida. Para esta gráfica, el personal del laboratorio buscara que, para cada posición del genoma, se obtenga la cobertura más alta posible. Esto eliminaria ambiguedad en los resultados y les daria una mejor confianza sobre la calidad de su secuenciación. Si se encuentran caidas muy pronunciadas y los datos muestras mayorvariación, podría significar una mala secuenciación o muestras contaminadas. Esto reduce la confiabilidad de los resultados.

CAPÍTULO 8

Discusión de resultados

Al estudiar los resultados obtenidos y Es evidente observar diferencias entre los resultados de linaje generados por cada uno de los tres modos de corrida. Particularmente, el modo Gencom no presenta resultados de linaje, o más bien devuelve un resultado indeterminado. Esto se puede atribuir a pangolín. Pangolin en línea de comando, posee dos diferentes motores de inferencia. El motor más novedoso es llamado Usher y el otro motor es el normal de Pangolin. Gencom, utiliza el Usher como motor de inferencia mientras que HAVoC y ViralFlow utilizan el motor normal. Por esta razón Gencom puede presentar diferentes resultados de linaje con diferentes muestras a los otros dos modos. También se observa una pequeña diferencia entre ViralFlow y HAVoC a pesar de usar el mismo motor de inferencia. HAVoC (al igual que Gencom) presenta como resultado Probable Ómicron y ViralFlow presenta Ómicron. Esto no causa mucho conflicto, solo señala una mayor cantidad de ambigüedad que se puede atribuir a los parámetros de Pangolin para cada modo Scher, 2022. Esto no afecta el resultado ni la interpretación del personal ya que se identifica el mismo linaje que es ampliamente reconocido como Ómicron.

Era importante para el éxito del proyecto que los archivos, una vez procesados estuvieran ordenados para poder ser accedidos fácilmente y que no se creara desorden dentro de la maquina donde el programa corre. En los tres modos de corrida se observa un orden diferente para almacenar los datos, pero en los tres se logra mantener un orden comprensible. En HAVoC y ViralFlow, se almacenan los datos generales en el directorio principal y los datos individuales para cada muestra dentro de su propio directorio. En Gencom, se almacenan los datos originales en un directorio y los resultados en otro en donde se dividen por tipo de archivo en vez de por muestra. Esto se puede observar en los resultados en las figuras 8, 9 y 10. Porsupuesto, el orden de los archivos y como se almacenan dentro de los diferentes directorios es subjetivo y cada usuario puede tener una preferencia diferente. Como se menciona en el análisis de resultados, estos datos se ordenan de manera similar y sistemática de modo que hay diferencias pero no es dificil encontrar los datos necesarios. Durante las pruebas con

usuarios, no se presentó nungún problema con encontrar los resultados o los datos deseados, por lo que se puede decir que el almacenamiento de estos datos es competente.

Se decidió no estandarizar el almacenamiento de archivos ya que hubiera agregado complejidad al programa y abierto más espacios para posibles fallos dado el numero potencial de diferentes archivos generados por cada modo de corrida. Las gráficas de cobertura solicitadas por el laboratorio para hacer control de calidad se generan con éxito. Se logró obtener una visualización satisfactoria y entendible para el personal de manera que puedan llevar a cabo un último control de calidad antes de obtener el resultado final que muestra la variante obtenida para cada muestra en un archivo csv. Gencom no presentó necesidad de generar las gráficas ya que genera las propias. Estas no se incluyen en resultados ya que no son un resultado del proyecto. Para ViralFlow y HAVoC se generaron los archivos de profundidad necesarios para el script de R utilizando samtools. Esta fue una adición simple pero de alto valor para el personal del Laboratorio ya que les permite revisar de otro modo, mas comprensible en algunas ocasiones, la calidad de sus muestras, datos y secuenciación.

Por último, se presentó el programa al personal del Laboratorio y se capacitó al equipo de secuenciación para utilizar el programa. Durante todo el proceso de desarrollo, se mantuvo comunicación frecuente con el equipó para que tuvieran una idea de como funcionaba el programa previo a su capacitación. Al finalizar el desarrollo, se tuvo una reunión con el equipo para mostrarles la funcionalidad completa e indicarles como ingresar el comando necesario junto sus parámetros. Se evaluó el uso del programa por parte de los miembros del equipo corriendo una prueba similar a la que se demostró, con datos reales en donde un miembro del equipo hizo uso del programa para llevar a cabo un análisis. Luego de esto, se dejó el programa finalizado en el laboratorio para su uso libre y habitual. No se han reportado errores, fallas en el programa o dificultad en el uso desde entonces. Se realizó una visita al laboratorio 3 meses después del deployment del programa para verificar su funcionalidad y el programa funcionó adecuadamente sin presentar ningún error.

En el laboratorio, se procesa una cantidad variante de muestras ya que depende de la disponibilidad de muestras, reactivos y tiempo. El número de muestras puede ser entre 48 y 96 cada 2 semanas a un mes. No obstante, se puede decir que el programa desarrollado tuvo un impacto positivo para el trabajo realizado en el laboratorio ya que se pueden obtener datos que son confiables y simples de obtener. Se conoce como estos datos son procesados y se puede verificar su exactitud al compararlos con datos generados por otros modos. También brinda la ventaja de que no se requiere de ninguna conexión a la red para que el programa funcione adecuadamente ya que pangolin realiza en análisis de manera local.

CAPÍTULO 9

Conclusiones

Se logró desarrollar un programa simple y funcional que realiza análisis genómico de SARS-CoV-2 a partir de datos generados de secuenciación. Este programa fue implementado desde el desarrollo hasta la capacitación de personal en el Laboratorio Nacional de Salud en Guatemala. El producto final presenta resultados de manera útil y ordenada. También se realizan controles de calidad tales como las gráficas de cobertura que son desplegados al usuario automáticamente a lo largo del proceso tales como gráficas de cobertura y fastqc. Este programa traerá beneficios para el personal del Laboratorio Nacional de Salud y les permitirá realizar control genómico local del virus SARS-CoV-2 y brindar información de relevancia a la población guatemalteca.

CAPÍTULO 10

Recomendaciones

Para futuras iteraciones del proyecto, se recomienda utilizar un software para construir pipelines como snakemake que permite crear analisis escalables y reproducibles (Mölder et al., 2021). De esta manera, se puede tener un proyecto más modular y ligeramente más amigable al usuario en cuanto su interfaz gráfica. Utilizar un software de este tipo también ayudaría a implementar paralelismo y reproducibilidad dentro del proyecto y podría facilitar la transferencia a otros equipos. También se recomienda trabajar usando variables de ambiente y directorios más generales para que el programa funcione sin importar la maquina en donde esté instalada o el directorio donde se almacenen los datos. Actualmente, los datos se deben almacenar en un directorio especifico dentro de la máquina. Se decidió realizar el programa de esta manera para no sacrificar el funcionamiento de las diversas pipelines. En lo particular, Gencom requería de una gran cantidad de directorios específicos. Adicionalmente, se consideró el orden en el que se almacenan los datos especificamente dentro del equipo utilizado por el LNS. El alcance de este proyecto comprendía una implementación específica para el LNS y no se extendía hacia otras posibles organizaciones o individuos. Por último, se recomienda buscar alternativas para estandarizar el almacenamiento de resultados y reducir el tiempo de corrida del programa, especialmente en modo gencom.

CAPÍTULO 11

Bibliografía

AAMCgenomics. (2022). *AAMCgenomics/gencom: Script for generation of a consensus genome annotation and associated plots*. https://github.com/AAMCgenomics/gencom

Anaconda Software Distribution. (2020). https://docs.anaconda.com/

Andrews, S. (2010). *FastQC: A Quality Control Tool for High Throughput Sequence Data*. http://www.bioinformatics.babraham.ac.uk/projects/fastqc/

CDC. (2020). *Coronavirus Disease 2019 (COVID-19)*. https://www.cdc.gov/coronavirus/2019-ncov/variants/genomic-surveillance.html

CDC. (2022). *How Flu Viruses Can Change: "Drift" and "Shift"*. https://www.cdc.gov/flu/about/viruses/change.htm

Conde, C. (2021). CIRCULAR No 10. Laboratorio Nacional de Salud. http://portal.lns.gob.gt/media/attachments/2021/04/09/alerta-epidemiologica-circular-10.pdf

Cucinotta, D., & Vanelli, M. (2020). WHO Declares COVID-19 a Pandemic. *Acta bio-medica : Atenei Parmensis*, *91*(1), 157-160. https://doi.org/https://doi.org/10.23750/abm.v91i1.9397

DataPipelines. (2021). *What is a Data Pipeline?* https://datapipelines.com/blog/what-is-a-data-pipeline/

de Salud, L. N. (2022). *Vigilancia Genómica de SARS-CoV-2*. http://portal.lns.gob.gt/index.php/component/sppagebuilder/62-secuencia-covid/

Dezordi, F., Neto, A., Campos, T., Jeronimo, P., Aksenen, C., Almeida, S., & Wallau, G. (2022). ViralFlow: A Versatile Automated Workflow for SARS-CoV-2 Genome Assembly, Lineage Assignment, Mutations and Intrahost Variant Detection. *Viruses*, *14*(2), 217. https://doi.org/https://doi.org/10.3390/v14020217

ecdc. (2022). *Methods for the detection and characterisation of SARS-CoV-2 variants - second update*. https://www.ecdc.europa.eu/en/publications-data/methods-detection-and-characterisation-sars-cov-2-variants-second-update

Ewels, P., Magnusson, M., Lundin, S., & Käller, M. (2016). MultiQC: Summarize analysis results for multiple tools and samples in a single report. *Bioinformatics*. https://doi.org/10.1093/bioinformatics/btw354

Fahrioğlu, U. (2018). Problems of unknown significance: Counseling in the era of next generation sequencing. *Balkan Journal of Medical Genetics*, *21*(1), 73-76. https://doi.org/https://doi.org/10.2478/bjmg-2018-0003

Gladman, S. (s.f.). *Introduction to de novo genome assembly for Illumina reads - Bioinformatics Documentation*. https://www.melbournebioinformatics.org.au/tutorials/tutorials/assembly/assembly-protocol/

Kalyanaraman, A., Hammond, K., Nieplocha, J., Krishnan, M., Palmer, B., Tipparaju, V., Harrison, R., Chavarria-Miranda, D., Makino, J., Bader, D., Cong, G., Hendrickson, B., Shalf, J., Donofrio, D., Rowen, C., Oliker, L., Wehner, M., & Gustafson, J. (2011). Genome Assembly. *Encyclopedia of Parallel Computing*, 755-768. https://doi.org/https://doi.org/10.1007/978-0-387-09766-4_402

KhanAcademy. (2016). *DNA Sequencing*. https://www.khanacademy.org/science/ap-biology/gene-expression-and-regulation/biotechnology/a/dna-sequencing

Leipzig, J. (2017). A review of bioinformatic pipeline frameworks. *Brief Bioinform*, *18*(3), 530-536. https://doi.org/10.1093/bib/bbw020

MERCK. (2022). *Sanger Sequencing Steps and Method*. https://www.sigmaaldrich.com/GT/es/technical-documents/protocol/genomics/sequencing/sanger-sequencing

Mölder, F., Jablonski, K., Letcher, B., Hall, M., Tomkins-Tinch, C., Sochat, J., V an Forster, Lee, S., Twardziok, S., Kanitz, A., Wilm, A., Holtgrewe, M., Rahmann, S., Nahnsen, S., & Köster, J. (2021). Sustainable data analysis with Snakemake. *F1000Res*, *10*(33).

Moore, S. (2021). *History of COVID-19*. https://www.news-medical.net/health/History-of-COVID-19.aspx#

MSPAS. (2022). *tablero COVID-19 Guatemala*. https://tablerocovid.mspas.gob.gt/tablerocovid/

NIH. (s.f.). *NCI Dictionary of Cancer Terms*. https://www.cancer.gov/publications/dictionaries/cancer-terms/def/sars-cov-2

OMS. (2021). *Tracking SARS-CoV-2 variants*. https://www.who.int/en/activities/tracking-SARS-CoV-2-variants/

Rambaut, A., Holmes, E., O'Toole, I., Hill, V., McCrone, J., Ruis, C., du Plessis, L., & Pybus, O. (2020). A dynamic nomenclature proposal for SARS-CoV-2 lineages to assist genomic epidemiology. *Nature Microbiology*, *5*(11), 1403-1407. https://doi.org/https://doi.org/10.1038/s41564-020-0770-5

Scher, E. (2022). *Usage*. https://cov-lineages.org/resources/pangolin/usage.html

Shah, D. (2021). *What is a variant? An expert explains*. https://wellcome.org/news/what-variant-expert-explains

ThomasJeffersonUniversity. (2020). *DNA and RNA*. https://cm.jefferson.edu/learn/dna-and-rna/

Truong Nguyen, P., Plyusnin, I., Sironen, T., Vapalahti, O., Kant, R., & Smura, T. (2021). HAVoC, a bioinformatic pipeline for reference-based consensus assembly and lineage assignment for SARS-CoV-2 sequences. *BMC Bioinformatics*, *22*(1). https://doi.org/https://doi.org/10.1186/s12859-021-04294-2

Wang, M.-Y., Zhao, R., Gao, L.-J., Gao, X.-F., Wang, D.-P., & Cao, J.-M. (2020). SARS-CoV-2: Structure, Biology, and Structure-Based Therapeutics Development. *Frontiers in Cellular and Infection Microbiology*, *10*. https://doi.org/10.3389/fcimb.2020.587269

Zhang, A. (2022). *How does Next Generation Sequencing work?* https://www.thetech.org/ask-a-geneticist/sanger-vs-next-gen-sequencing

Zhu, H., Wei, L., & Niu, P. (2020). The novel coronavirus outbreak in Wuhan, China. *BMC Bioinformatics*, *5*(6). https://doi.org/https://doi.org/10.1186/s41256-020-00135-6

CAPÍTULO 12

Anexos

12.1. Diagramas de flujo de algoritmos utilizados

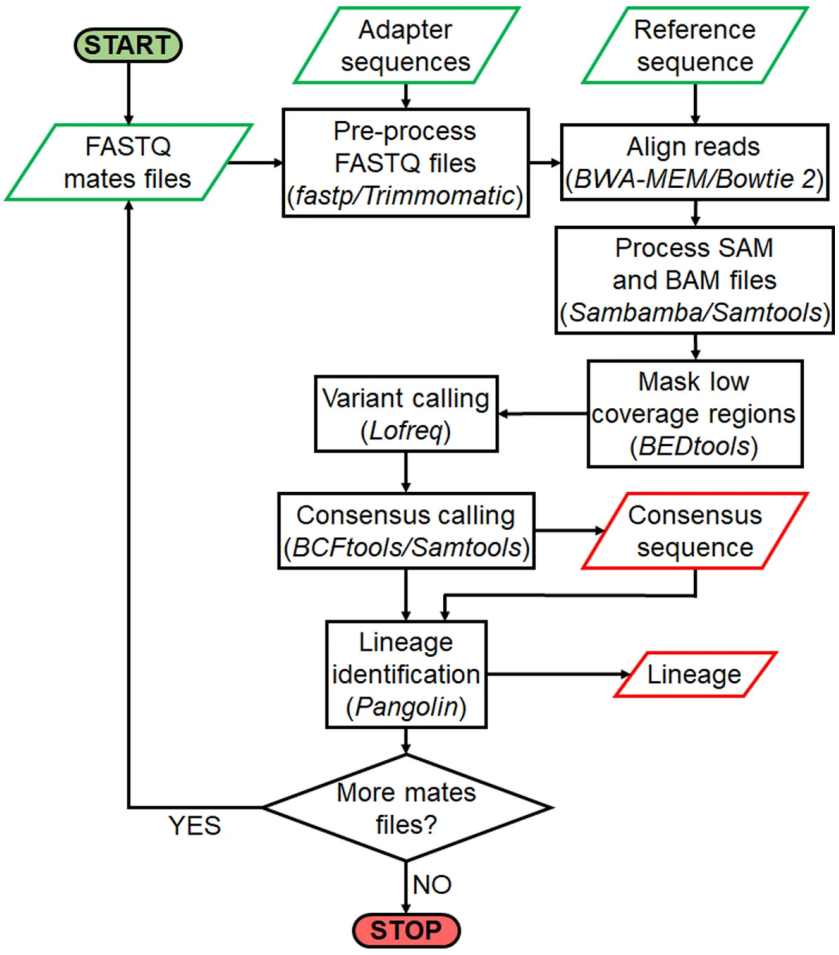


Figura 12: Diagrama de Flujo del algoritmo de HAVoC

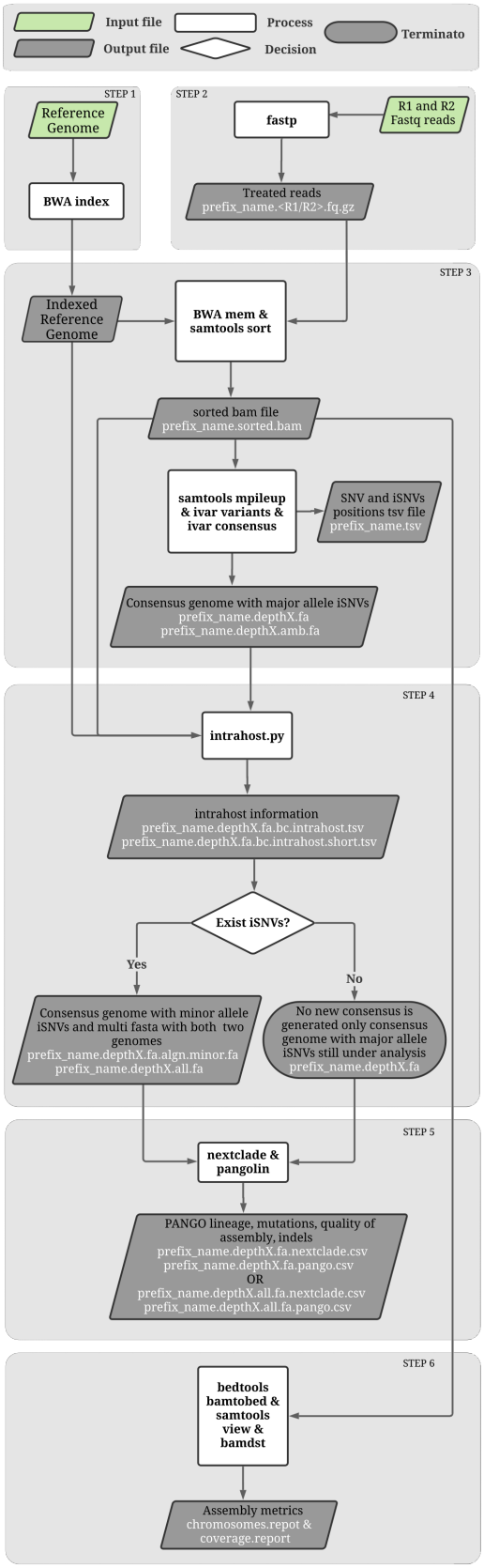


Figura 13: Diagrama de Flujo del algoritmo de ViralFlow

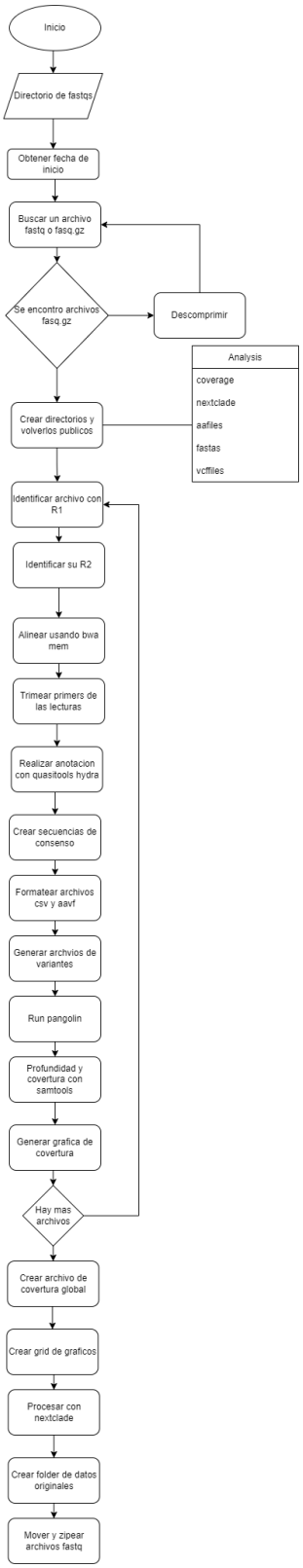


Figura 14: Diagrama de Flujo del algoritmo de Gencom

Figura 15: Diagrama de Flujo del algoritmo de COVLNS.sh (el programa principal)

12.2. Repositorio en Github del Proyecto:

https://github.com/Estdv/COVLNS

12.3. Versiones de softwares empleados:

- ViralFlow v.0.0.6
- HAVoC v1
- Gencom v1
- Docker v20.10
- Samtools v1.16.1
- pdfunite v0.6.0
- fastqc v0.11.9
- multiqc v1.13
- R v4.2.2

12.4. Guia de uso:

Descargar e instalar requerimientos listados en el repositorio y en la sección 12.3

Descargar archivos y correr el siguiente comando:

bash COVLNS.sh [path] [mode]

path: Directorio en donde se encuentran los archivos fastq

mode: software de variantes utilizado [(H,h):HAVoC, (V,v):Viralflow, (G,g):Gencom] (default: h)

ex: bash COVLNS.sh /home/fastqs/ h

Nota: será necesario hacer varias modificaciones en el codigo en cuanto a los directorios para que el programa funcione adecuadamente

12.5. Capturas de pantalla del programa en ejecución

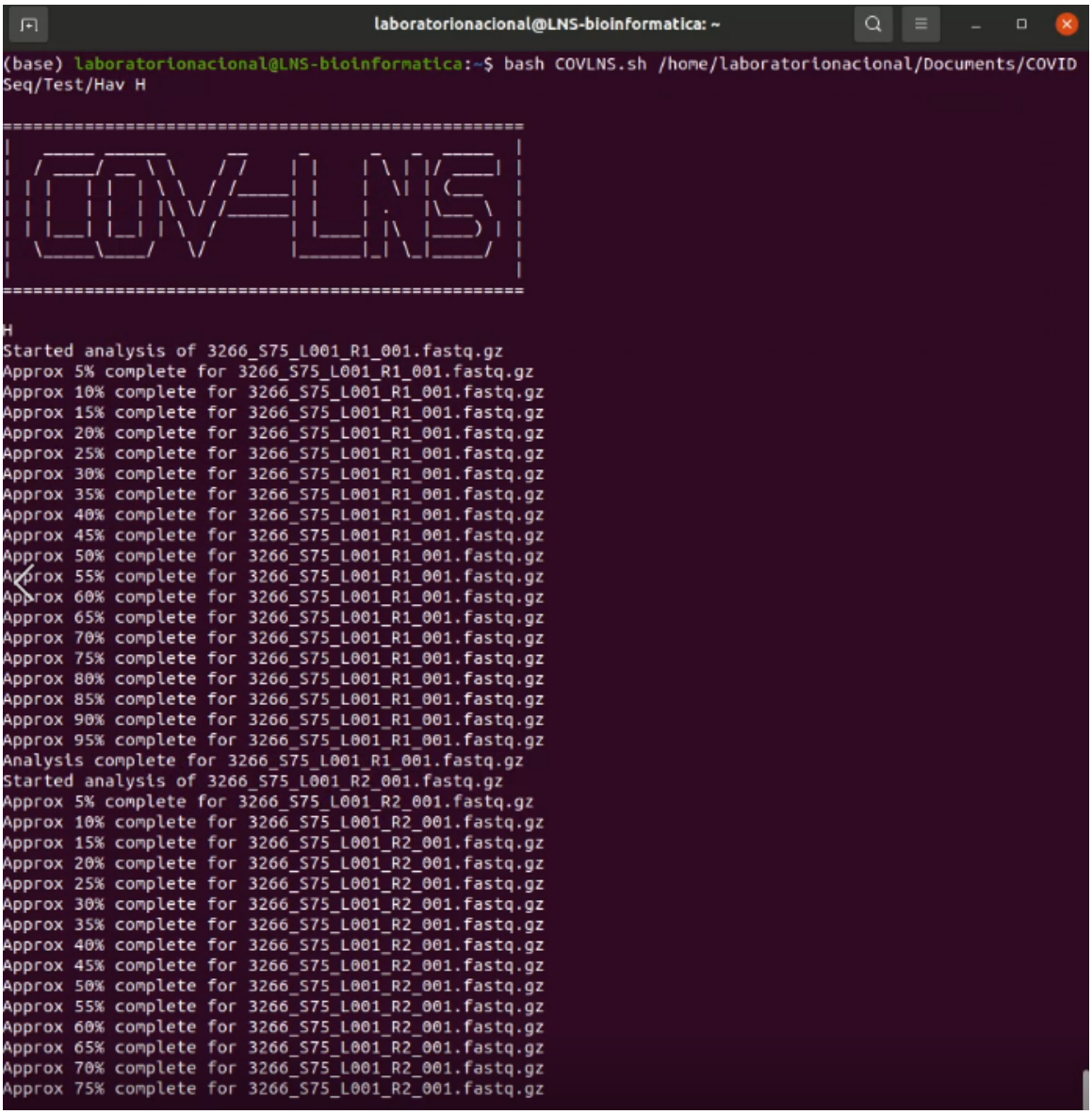


Figura 16: COVLNS siendo ejecutado en modo HAVoC. En la parte superior se ve el comando utilizado, luego un mensaje de bienvenida en formato ascii, luego una letra H para verificar el modo y por último se puede ver el inicio del control de calidad realizado por fastqc

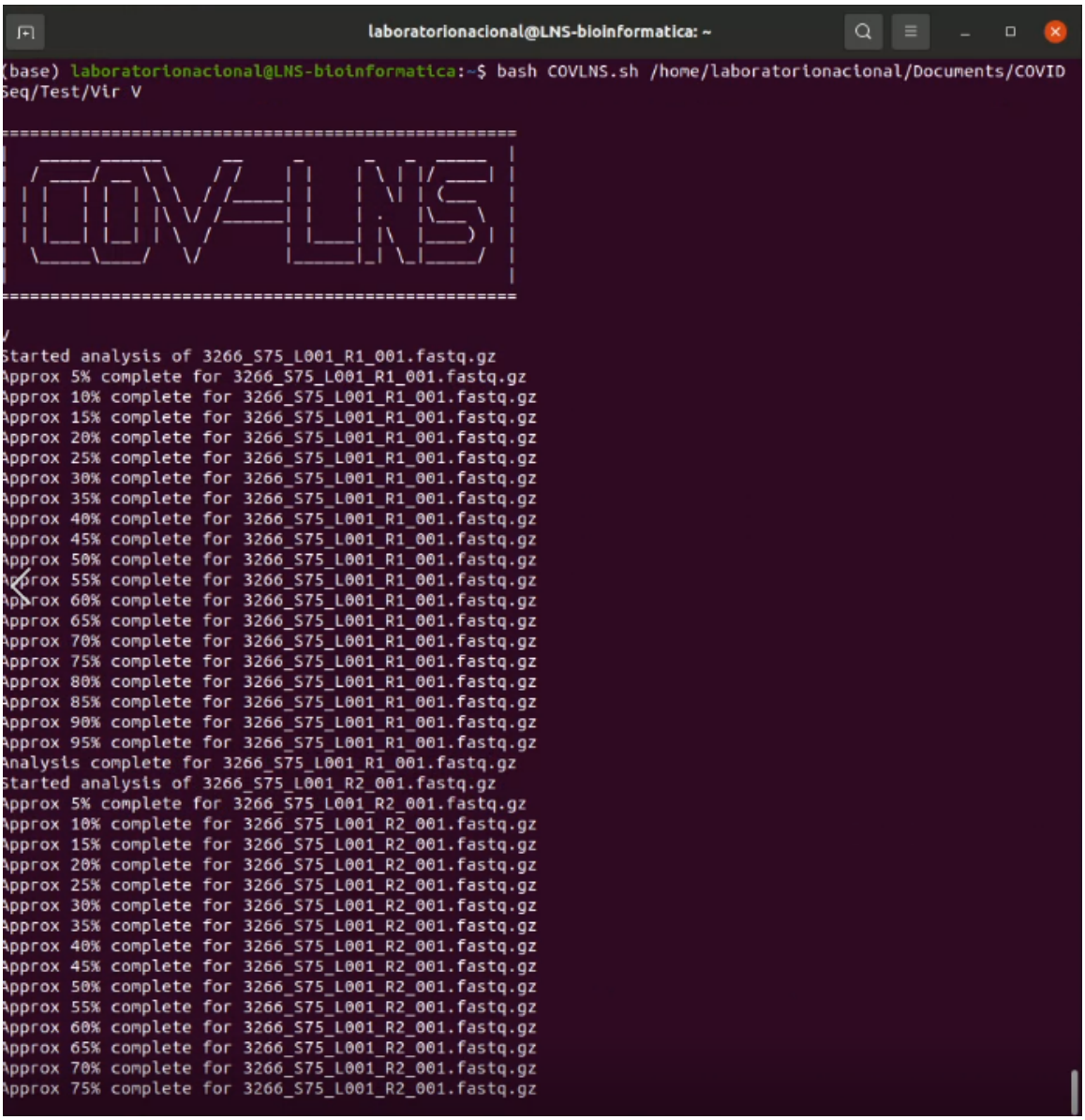


Figura 17: COVLNS siendo ejecutado en modo ViralFlow. En la parte superior se ve el comando utilizado, luego un mensaje de bienvenida en formato ascii, luego una letra V para verificar el modo y por último se puede ver el inicio del control de calidad realizado por fastqc

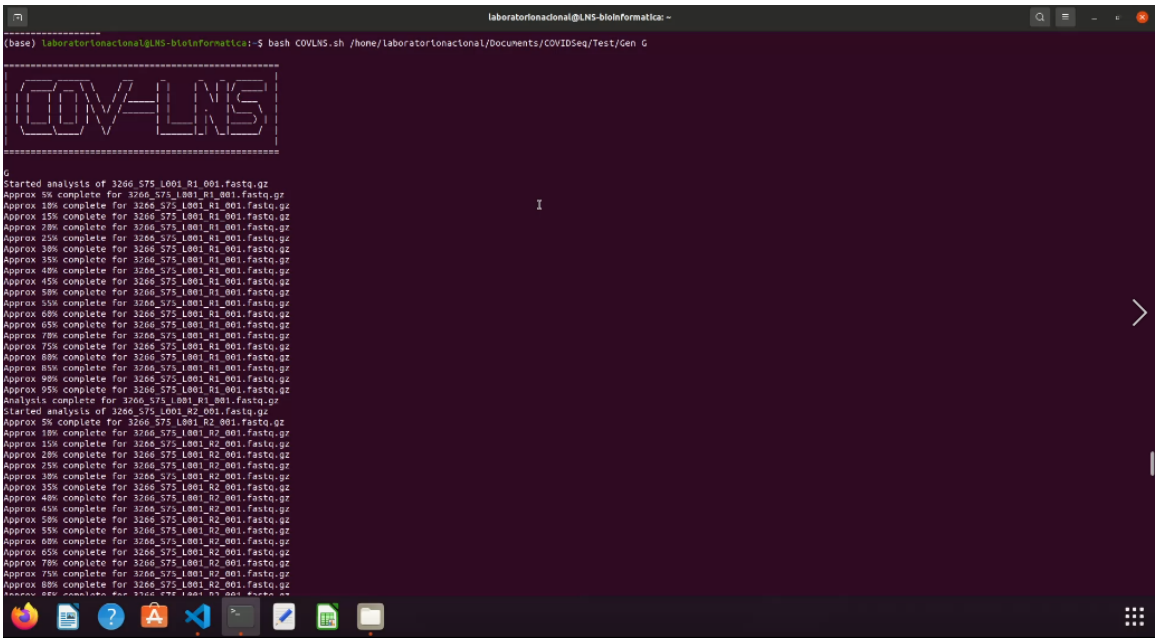


Figura 18: COVLNS siendo ejecutado en modo Gencom. En la parte superior se ve el comando utilizado, luego un mensaje de bienvenida en formato ascii, luego una letra G para verificar el modo y, por último, se puede ver el inicio del control de calidad realizado por fastqc

12.6. Tablas con datos usados como ejemplo

Las tablas han sido sido divididas para que se puedan desplegar todos los datos

Cuadro 1: Tabla con datos utilizados en resultados como ejemplo para HAVoC

taxon	lineage	conflict	ambiguity_score	scorpio_call	scorpio_support	note
3266	BA.1	0.57	0.89	Probable Omicron (BA.1-like)	0.66	
3271	BA.1	0.61	0.92	Probable Omicron (BA.1-like)	0.68	

scorpio_conflict	scorpio_notes	version
0	scorpio call: Alt alleles 39; Ref alleles 0; Amb alleles 18; Oth alleles 2	PLEARN-v1.8
0	scorpio call: Alt alleles 40; Ref alleles 0; Amb alleles 17; Oth alleles 2	PLEARN-v1.8

pangolin_version	scorpio_version	constellation_version	is_designated	qc_status
4.0.6	0.3.17	v0.1.10	FALSE	pass
4.0.6	0.3.17	v0.1.10	FALSE	pass

qc_notes	note
Ambiguous_content:0.07	
Ambiguous_content:0.06	

Cuadro 2: Tabla con datos utilizados en resultados como ejemplo para ViralFlow

taxon	lineage	conflict	ambiguity_score	scorpio_call	scorpio_support
3266_S75_L001	BA.1.1	0	0.963491	Omicron (BA.1-like)	0.7119
3266_S75_L001_minor	BA.1.1	0	0.963491	Omicron (BA.1-like)	0.661
3271_S76_L001	BA.1.1	0	0.993919	Omicron (BA.1-like)	0.9322
3271_S76_L001_minor	BA.1.1	0	0.993919	Omicron (BA.1-like)	0.8644

scorpio_conflict	version	pangolin_version	pangoLEARN_version	pango_version
0	PLEARN-v1.2.124	3.1.20	2/2/2022	v1.2.124
0.0339	PLEARN-v1.2.124	3.1.20	2/2/2022	v1.2.124
0	PLEARN-v1.2.124	3.1.20	2/2/2022	v1.2.124
0.0508	PLEARN-v1.2.124	3.1.20	2/2/2022	v1.2.124

status	note
passed_qc	scorpio call: Alt alleles 42; Ref alleles 0; Amb alleles 14; Oth alleles 3
passed_qc	scorpio call: Alt alleles 39; Ref alleles 2; Amb alleles 13; Oth alleles 5
passed_qc	scorpio call: Alt alleles 55; Ref alleles 0; Amb alleles 1; Oth alleles 3
passed_qc	scorpio call: Alt alleles 51; Ref alleles 3; Amb alleles 1; Oth alleles 4

Cuadro 3: Tabla con datos utilizados en resultados como ejemplo para Gencom

taxon	lineage	conflict	ambiguity_score	scorpio_call	scorpio_support
3266	Unassigned	0		Probable Omicron (Unassigned)	0.7
3271	Unassigned	0.5		Probable Omicron (Unassigned)	0.73

scorpio_conflict	scorpio_notes	version
0.21	scorpio call: Alt alleles 23; Ref alleles 7; Amb alleles 3; Oth alleles 0	PUSHER-v1.8
0.27	scorpio call: Alt alleles 24; Ref alleles 9; Amb alleles 0; Oth alleles 0	PUSHER-v1.8

pangolin_version	scorpio_version	constellation_version	is_designated	qc_status
4.0.6	0.3.17	v0.1.10	FALSE	pass
4.0.6	0.3.17	v0.1.10	FALSE	pass

note
Usher placements: BA.1.1(1/1); scorpio replaced lineage inference BA.1.1
Usher placements: BA.1.1(1/2) BA.1.1.15(1/2); scorpio replaced lineage inference BA.1.1.15