Daniel Felipe Bautista Carrizosa

dbautistac@unimagdalena.edu.co

Práctica control de calidad de lecturas de ONT con pycoQC

Uno de los procedimientos clave para cualquier tipo de análisis bioinformático es realizar un control de calidad de nuestras muestras, asegurándonos que no vamos a introducir datos poco confiables que nos pueden llevar a conclusiones erradas. En nuestro caso, explorar las métricas de nuestro experimento y eliminar las secuencias que poseen un alto porcentaje de error de secuenciación es un paso importante para obtener genomas de SARS-CoV-2 de alta calidad.

Existen varios programas que permiten visualizar de manera global los resultados de la secuenciación, midiendo diferentes métricas que nos hablan de la calidad de la secuenciación, como la mediana, el mínimo y el máximo de la longitud de las lecturas, el aspecto que tienen la calidad media y la distribución de la calidad de nuestra secuenciación, entre otros.

Muchos de los programas utilizados, como FASTQC, están pensados para manejar datos de Sanger o Illumina, que poseen diferentes características que las generadas por instrumentos de lecturas largas. Es por esto que para esta práctica vamos a usar el programa pycoQC, diseñado específicamente para Nanopore, que genera reportes interactivos que podemos explorar en nuestro navegador.

1. Preparación del material

Diríjase al directorio donde guardamos los resultados del llamado de bases con Guppy. Aunque los datos importantes para seguir con la metodología del curso son los archivos FASTQ que se encuentran en la carpeta `pass`, existe un documento llamado `sequencing_summary.txt` que contiene un registro de todas las lecturas procesadas en el llamado de bases. En este archivo se encuentran puntajes de calidad para cada base (como en FASTQ), pero tienen otro tipo de información:

```
filename     read_id run_id  batch_id     channel mux start_time
duration       num_events   passes_filtering       template_start
num_events_template template_duration    sequence_length_template
mean_qscore_template     strand_score_template    median_template
mad_template
```

Todos estos campos nos arrojan mayor información sobre la corrida. Pronto vamos a ver qué nos puede mostrar estos datos.

2. Uso de pycoQC

Para correr el programa, usamos el comando `pycoQC.` Esto nos arroja las opciones posibles. Si queremos conocer la versión del programa podemos correr el comando:

```
pycoQC --version
```

Para usar pycoQC necesitamos señalar el archivo de `sequencing_summary.txt` de nuestra corrida e indicar cómo queremos guardar el archivo de salida. El comando podría ser:

```
pycoQC –f Set_fastq/sequencing_summary.txt –o practica.html
```

El cual nos genera el archivo `practica.html` en el directorio donde nos encontremos. Para poder visualizar el archivo en un entorno gráfico, es necesario abrirlo desde el navegador.

Explore cuidadosamente los resultados. ¿Qué cosas llaman la atención? ¿Puede observar una relación entre la longitud de las lecturas y la calidad de las bases? ¿Cómo cambió la secuenciación a lo largo del experimento?