Instituto Nacional de Medicina Genómica

Aprende a utilizar los flujos de análisis bioinformáticos internos del Inmegen

Introducción a NextFlow y Docker

Profesores: Dra. Alejandra Cervera
Dra. Laura Gómez
Dr. Daniel Pérez

nextflow

Temario del curso

Instituto Nacional de Medicina Genómica

- Repaso de Bash
- Introducción a Nextflow y Docker
- Cuantificación y análisis de expresión diferencial
- Identificación automatizada de variantes germinales
- Identificación automatizada de variantes somáticas
- Identificación automatizada de variantes de datos de RNA-seq

Instituto Nacional de Medicina Genómica

Repaso de Unix

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Rutas en Linux - Absolutas vs. Relativas -

- **Ruta absoluta:** empieza desde la raíz del sistema de archivos (`/`). Ejemplo: `/home/usuario/proyecto`.
- **Ruta relativa:** se basa en la ubicación actual del usuario. Usa `./` para referirse a la carpeta actual y `../` para subir un nivel en el directorio. Ejemplo: `../proyecto`.

Comandos comunes para manejar rutas:

- `pwd`: muestra el directorio actual.
- `cd`: cambia de directorio.
  - `cd /ruta/absoluta` cambia a una ruta absoluta
  - `cd ..` sube un nivel en el directorio.
- `ls`: lista los archivos y directorios en la ubicación actual.

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Copiar Archivos y Directorios (comando `cp`):

- **Sintaxis básica:** `cp archivo_origen archivo_destino`

Ejemplo: `cp archivo.txt copia_archivo.txt` (crea una copia de `archivo.txt` con el nombre `copia_archivo.txt`)

Copiar directorios completos: Opción **-r** para copiar directorios de manera recursiva.

Ejemplo: `cp -r carpeta_origen carpeta_destino`

Opciones útiles:

`-i`: pregunta antes de sobrescribir archivos.

`-u`: copia solo cuando el archivo origen es más reciente que el archivo destino o cuando el archivo destino no existe.

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Mover Archivos y Directorios (comando `mv`):

- **Sintaxis básica:** `mv archivo_origen archivo_destino`

Ejemplo: `mv archivo.txt nueva_ubicacion/` (mueve `archivo.txt` a `nueva_ubicacion/`)

También se puede usar `mv` para renombrar un archivo:

```
mv archivo.txt nuevo_nombre.txt
```

Opciones útiles:

`-i`: pregunta antes de sobrescribir archivos.

`-u`: mueve solo cuando el archivo origen es más reciente que el archivo destino o cuando el archivo destino no existe.

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Modificación de permisos con `chmod`:

- **Uso simbólico:** `chmod u+x archivo.txt` (añade permiso de ejecución para el usuario).
- **Uso numérico:** `chmod 755 archivo.txt`
  4 para lectura, 2 para escritura y 1 para ejecución.

  propietario, grupo, otros

Cambio de propietario y grupo con `chown`:

Ejemplo: `chown usuario:grupo archivo.txt`

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Grupos en Linux

Los grupos permiten organizar usuarios para administrar permisos y acceso a recursos compartidos, cada usuario puede pertenecer a uno o más grupos.

Comandos comunes para manejar grupos:

`groups`: muestra los grupos a los que pertenece un usuario.

IMPORTANTE: Para utilizar los flujos de Inmegen debes de pertenecer al grupo docker:

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Conexión a un servidor mediante `SSH`

Concepto de `SSH` (Secure Shell).

Es un protocolo de red que permite la comunicación segura entre dos sistemas a través de una red insegura.

`SSH` es comúnmente usado para acceder a servidores remotos de manera segura.

Uso del comando `SSH`:

Sintaxis básica: `ssh usuario@dirección_ip`

Ejemplo: `ssh alumnoN@10.0.15.11`

Opciones más comunes:

`-p`: especifica un puerto diferente si el servidor SSH no usa el puerto por defecto (22).

`-i`: indica el archivo de la clave privada para autenticarse.

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Copiado de archivos con `SCP` (Secure Copy Protocol)

- **Concepto de `SCP`**:
  - `SCP` permite copiar archivos de manera segura entre sistemas locales y remotos utilizando SSH.
- **Uso del comando SCP**:
  - Sintaxis básica para copiar un archivo desde el sistema local al servidor remoto:
    - `scp archivo.txt usuario@dirección_ip:/ruta/de/destino`
  - Copiar un archivo desde el servidor remoto al sistema local:
    - `scp usuario@dirección_ip:/ruta/archivo.txt /ruta/local/destino`
  - Copiar directorios completos usando la opción `-r`:
    - `scp -r directorio usuario@dirección_ip:/ruta/de/destino`
- **Opciones útiles**:
  - Uso de `-P` para especificar un puerto diferente.
  - Uso de `-i` para indicar una clave privada.

Conceptos básicos de bash

Instituto Nacional de Medicina Genómica

Editor de texto Nano:

- **Nano** es un editor de texto simple y fácil de usar que se utiliza en la línea de comandos de Linux.
- Ideal para principiantes y para ediciones rápidas de archivos de configuración y scripts.

Comandos básicos en Nano:

- **Abrir un archivo**: `nano archivo.txt` : si el archivo no existe, Nano creará uno nuevo.
- **Guardar cambios**:
  - `Ctrl + O` y luego `Enter` para guardar el archivo.
- **Salir del editor**:
  - `Ctrl + X` para salir de Nano. Si no has guardado los cambios, te pedirá confirmación.

Atajos útiles en Nano:

- `Ctrl + K`: corta una línea.
- `Ctrl + U`: pega la línea cortada.
- `Ctrl + W`: busca texto dentro del archivo.
- `Ctrl + G`: muestra la ayuda de Nano.
  -

Instituto Nacional de Medicina Genómica

Introducción a Nextflow y Docker

Introducción a Nextflow

Instituto Nacional de Medicina Genómica

¿Qué es **Nextflow**?

Nextflow es una herramienta para el desarrollo y la automatización de flujos de trabajo científicos y bioinformáticos.

Permite diseñar y ejecutar flujos de trabajo (*pipelines*) de análisis de datos de manera eficiente y reproducible.

Es compatible con herramientas y lenguajes de programación populares, como: Bash, Python y R.

Introducción a Nextflow

Instituto Nacional de Medicina Genómica

Características principales de **Nextflow**

- **Paralelización:** Nextflow puede ejecutar tareas en paralelo, aprovechando al máximo los recursos de cómputo disponibles.
- **Portabilidad:** Los flujos de trabajo se pueden correr en diferentes entornos de cómputo, como una computadora local, un cluster de supercómputo o la nube, usando tecnologías como Docker y Singularity.
- **Reproducibilidad:** Los pipelines de Nextflow aseguran que los análisis sean reproducibles, registrando las versiones de los datos y las herramientas utilizadas.

Introducción a Nextflow

Instituto Nacional de Medicina Genómica

Conceptos básicos de **Nextflow**

- **Pipeline:** Es una serie de pasos (procesos) que transforman los datos de entrada en resultados de salida.
- **Proceso:** La unidad básica de trabajo en Nextflow. Cada proceso tiene una entrada, un script (lo que ejecuta), y una salida.
- **Canales:** Son los conectores que permiten que los datos fluyan entre los diferentes procesos dentro de un pipeline. Los canales definen cómo se pasa la información de una tarea a la siguiente.

Introducción a Nextflow

Instituto Nacional de Medicina Genómica

Sintaxis básica

- Los flujos de trabajo en Nextflow se escriben en un lenguaje de scripts basado en **Groovy**, lo que los hace fáciles de entender para personas con conocimientos en Java y Bash.

```
nextflow.enable.dsl=2

// Importa los módulos de procesos desde otro archivo (opcional, pero recomendado en DSL 2
include { processA; processB } from './modules/my_processes.nf'

workflow {
    // Define los datos de entrada
    params.samples = ['sample1', 'sample2', 'sample3']

    // Define el canal de entrada que alimentará al primer proceso
    Channel
        .from(params.samples)
        .set { inputSamples }

    // Ejecuta el proceso A usando el canal de entrada
    processA(inputSamples)
        .set { processedData }

    // Ejecuta el proceso B con los datos procesados del proceso A
    processB(processedData)
}
```

```
// Define el primer proceso: processA
process processA {
    input:
    val sample

    output:
    file("output_${sample}.txt")

    script:
    """
    echo "Processing ${sample}" > output_${sample}.txt
    """
}

// Define el segundo proceso: processB
process processB {
    input:
    file processedFile

    output:
    file("final_result.txt")

    script:
    """
    echo "Finalizing analysis for ${processedFile}" > final_result.txt
    """
}
```

Introducción a Docker

Instituto Nacional de Medicina Genómica

¿Qué es Docker?

- **Docker** es una plataforma de software que permite crear, ejecutar y gestionar aplicaciones dentro de contenedores. Un **contenedor** es una unidad estandarizada que empaqueta una aplicación y todas sus dependencias para que se ejecute de manera consistente en cualquier entorno.

Componentes clave de Docker

- **Imagen:** Una imagen de Docker es un archivo compacto que contiene todo lo necesario para ejecutar una aplicación (código, dependencias, configuración). Las imágenes se utilizan para crear contenedores.
- **Contenedor:** Un contenedor es una instancia en ejecución de una imagen. Se puede crear, iniciar, detener y eliminar según sea necesario.
- **Dockerfile:** Un archivo de texto que contiene una serie de instrucciones para crear una imagen de Docker. Define cómo debe construirse la imagen y qué debe incluir.
- **Docker Hub:** Un repositorio público donde los desarrolladores pueden almacenar y compartir imágenes de Docker.

Introducción a Docker

Instituto Nacional de Medicina Genómica

Ejemplo de Dockerfile:

```
# Usa una imagen base de Python oficial
FROM python:3.9-slim

# Establece el directorio de trabajo dentro del contenedor
WORKDIR /app

# Copia el archivo de requerimientos a la imagen
COPY requirements.txt .

# Instala las dependencias necesarias
RUN pip install --no-cache-dir -r requirements.txt

# Copia el resto del código de la aplicación al contenedor
COPY . .

# Especifica el comando que se ejecutará al iniciar el contenedor
CMD ["python", "app.py"]
```

- `FROM python:3.9-slim` : Define la imagen base. En este caso, se usa una versión ligera de Python 3.9.
- `WORKDIR /app`: Establece `/app` como el directorio de trabajo donde se ejecutarán los comandos posteriores.
- `COPY requirements.txt .` : Copia el archivo `requirements.txt` desde tu máquina local al directorio de trabajo en el contenedor.
- `RUN pip install --no-cache-dir -r requirements.txt` : Instala las dependencias listadas en `requirements.txt` utilizando `pip`.
- `COPY . .`: Copia todo el contenido del directorio actual (código fuente) al directorio de trabajo del contenedor.
- `CMD ["python", "app.py"]` : Define el comando que se ejecutará cuando se inicie el contenedor, en este caso, ejecutará el archivo `app.py`

Introducción a Docker

Instituto Nacional de Medicina Genómica

Comandos más comunes de Docker

- **`docker build`**: crea una imagen a partir de un Dockerfile
  - `docker build -t nombre-imagen .`
- **`docker run`**: ejecuta un contenedor basado en una imagen.
  - `docker run -v /ruta/local:/ruta/contenedor nombre-imagen`
- **`docker images`**: lista todas las imágenes en tu máquina local.
- **`docker rmi nombre-imagen`**: elimina una imagen.
- **`docker ps`**: muestra los contenedores en ejecución.
- **`docker stop nombre-contenedor`**: detiene un contenedor en ejecución.
- **`docker rm nombre-contenedor`**: elimina un contenedor.

Ejercicios

1. Automatiza el análisis de calidad con fastqc
2. Automatiza el recorte de adaptadores y bases de mala calidad con fastp
3. Automatiza el resumen del análisis con multiqc

Repositorio del curso

Instituto Nacional de Medicina Genómica

Repositorios e imagen de docker: `pipelinesinmegen`

¿Dudas?