

Informe Laboratorio 7. Usando Docker en Bioinformática

Este laboratorio ha resultado especialmente ameno e interesante al representar un cambio significativo respecto a la dinámica de los laboratorios anteriores. Hasta ahora, las prácticas seguían un patrón similar centrado en optimización de código, mientras que este laboratorio introduce conceptos nuevos. Al inicio, conceptos como imagen, contenedor o Dockerfile resultaban abstractos y algo confusos. Sin embargo, la estructura del laboratorio facilita enormemente la comprensión progresiva de estos elementos. Las clases teóricas y la lectura detallada de la teoría del tema (*hpc-docker.pdf*), junto con los recursos externos proporcionados (especialmente las guías de buenas prácticas), han sido fundamentales para entender cada paso del proceso.

Un aspecto particularmente destacable es la claridad en la redacción y estructura del enunciado. Cada comando está explicado con suficiente detalle para entender su función. La progresión desde conceptos básicos (ejecutar contenedores preexistentes con `hello-world`) hasta tareas más complejas (crear imágenes propias interactivamente y mediante Dockerfile) está muy bien diseñada. Esta gradualidad permite asimilar cada concepto antes de pasar al siguiente. Las notas al pie de página merecen mención especial. A lo largo del documento, estas aclaraciones resuelven dudas anticipadamente y previenen errores comunes. Estas notas demuestran una comprensión clara de los puntos donde los estudiantes suelen encontrar dificultades, ahorrando considerable tiempo y frustración.

La aplicación práctica con BLAST ha sido muy ilustrativa. Utilizar una herramienta bioinformática real en un contenedor muestra claramente la utilidad de Docker. El ejemplo con la base de datos de *zebrafish* y secuencias de priones humanos conecta directamente con aplicaciones reales de bioinformática, haciendo el laboratorio más relevante.

La actividad extra de crear un Dockerfile resulta especialmente valiosa. Aunque crear imágenes interactivamente (apartado 3.3) es más intuitivo inicialmente, el uso de Dockerfiles representa la forma profesional y reproducible de trabajar con contenedores.

En cuanto a aspectos mejorables, existe una pequeña inconsistencia en la nomenclatura del notebook de Python a utilizar. El enunciado hace referencia a *reduc-operation-array-login.ipynb*, mientras que en el laboratorio de Python avanzado el archivo incluía la partícula "-par" en su nombre (*reduc-operation-array-par-login.ipynb*). Aunque esta diferencia es menor y se resuelve fácilmente revisando el material previo, sería conveniente especificar el nombre exacto del archivo para evitar confusiones innecesarias.

Otro aspecto a considerar es que el laboratorio presenta tres formas diferentes de interactuar con Docker: línea de comandos, interfaz gráfica de Docker Desktop, y Docker Hub web. Si bien esto ofrece flexibilidad, podría resultar útil incluir una breve orientación sobre cuándo es más apropiado utilizar cada interfaz.

En conclusión, la combinación de claridad en las instrucciones, progresión lógica de dificultad, ejemplos prácticos relevantes y documentación de buenas prácticas convierten este laboratorio en una experiencia de aprendizaje muy completa. Las competencias adquiridas sobre creación y gestión de contenedores Docker son directamente aplicables a proyectos reales de investigación bioinformática, donde la reproducibilidad y la gestión de entornos computacionales son aspectos críticos.