

Informe laboratorio 5: Control de versiones con Git y GitHub

Isabel María Fernández Lizán

Explicación ficheros adjuntos:

- ***cut_files.sh***: este programa va a cortar los ficheros *.fastq* en partes iguales y va a nombrar a primera porción cortada como *SampleX.00*, donde *X* es el número de muestra (1,2,3 o 4). Este programa consta de los siguientes parámetros:
`cut_files.sh [n_partes] [fichero]`
donde *n_partes* es el número de partes que queremos que se corte el fichero (en mi caso 14) y *fichero* es el fichero que queremos que corte (*Sample1.fastq*, *Sample2.fastq*, etc).
- ***submit_lab5_pacioli-alumno14.sh***: lanza el anterior programa a pacioli para que se ejecute sobre los cuatro ficheros *.fastq*. Con el argumento `--ntasks-per-node=4` y al añadir `&` en el bucle indicamos que el programa se ejecute en paralelo usando 4 núcleos (un núcleo por fichero) y de esta forma reducimos el tiempo en el que obtenemos los resultados. Para ver lo que tarda en ejecutarse el programa he añadido `time wait` al final del bucle.
Por último, se reemplaza la extensión *.00* por *.fastq* para que la primera porción cortada se llame igual que su fichero de procedencia.

Aspectos positivos y a mejorar de la práctica:

Gracias a esta práctica he aprendido a manejar por mi repositorio local y por el remoto, además de realizar el control de versiones. Me ha facilitado el entendimiento de los principales comandos que se utilizan en *git* ya que te va explicando poco a poco para que sirva cada uno. Con el *Shell script* he ideado una estrategia para aprovechar los núcleos disponibles en el clúster y para trabajar en paralelo, por lo que me ha servido para fijar lo aprendido con la práctica 2. Como podemos ver en *slurm-23426*, usando los cuatro núcleos obtenemos los resultados en tan solo 10 milisegundos.

He echado en falta un pequeño apartado que te permita ver las diferencias entre las distintas versiones con el comando *git diff* o, por ejemplo, borrar la última versión en la que se hace *commit*. Como comenté en clase, también sería interesante hacer una colaboración en GitHub con mis compañeros, ya que, en mi opinión, es uno de los fuertes de esta plataforma (junto al control de versiones).