WORKSHOP GITHUB (CIENCIA REPRODUCIBLE)

Git: sistema de control de versiones distribuido, permite traquear lo que se va haciendo en un proyecto. En cada commit puedes ver en que parte del proyecto se han hecho los cambios y te dicen que líneas de código que se cambiaron.

Github: es una nube que integra Git.

Hay que tener cuidado cuando instalas git de tener Rstudio cerrado sino no lo reconoce.

Hay que instalar el Token de Git (en mi caso he puesto caducidad una semana seguramente me de problema mas adelante, debería poner sin caducidad alomejor)

Para empezar a trabajar con Git y Rstudio hay que conectar ambos con una carpeta. Para eso primero debemos crear un proyecto. Primero debemos tener un repositorio creado en Github y copiar la URL de ese repositorio. Para crear proyecto debemos clicar en versión control y meter la URl del repositorio. Una vez este todo, debe aparecer una pestaña de git en Rstudio.

El archivo de gitignore es un archivo txt que no queremos que git no rastree. Este archivo no se sube y se baja. Tu tienes por ejemplo tus datos en local y los lees como gitignore y no se están subiendo y bajando todo el rato, así se aligera. Para ello desde la pestaña de git le damos click derecho a lo que queramos ignorar. OJO: las carpetas vacías no las een y por tanto no se pueden ignorar.

El commit se debe hacer cuando acabes la sesión si es algo rutinario. En cambio, si estas haciendo cosas importantes se debe hacer en cuanto tengas pequeños pasos que sepas que funcionan.

Puedes abrir el historial de versiones (todos los commit hechos) en Rstudio y acceder exactamente al momento del código que tu quieres.

Si nos metemos en un código dentro del repositorio de GitHub podemos pinchar en la pestañita que pone Blame y ahí se puede ver quien ha hecho cada cambio dentro del código. Además si vemos en que línea esta el error, se le puede dar a botón derecho y darle a permalink y mandarle a la otra persona justo la línea de código en la que se ha equivocado.

Para unir ramas siempre hay que hacerlo desde la rama a la que quieres unirte. Github te identifica que hay dos procesos difereentes y con un pull request se unen ambas ramas.

QUARTO:

Sirve para crear documentos desde R. (es como r makdown pero mejon)

Es muy importante tener en cuenta los espacios cuando escribimos el primer yam (donde pone toda la información del titulo autores etc)

Kable y el paquete es knitr y otra para mas edicion es Kableextra