Nos va a interesar mirar siempre como desarrollador los datos de (y en) producción, mejor que los datos de testing (entorno no productivo), porque son datos reales, aunque a veces no será posible. Los notebooks son para buscar y mirar datos para extraer conclusiones, relaciones, hacer regresiones, etc., pero no para modificar nada. ¿Cómo? Con un lenguaje de programación como Python o R. En el notebook hay parte de código y parte de comentarios (Markdown). Kernel es el núcleo.

Párrafo o celda es cada parte de código que ejecutamos. Los notebooks tienen un orden entre las instrucciones y están relacionadas. Si reinicio el Kernel se borrarán las variables previamente introducidas.

**Google Colab** (notebook más usado entre Data Scientists): para ejecutar Mayus + Enter. Le das a conectar para que te conecte con una VM.

Entornos de ejecución: cambiar el entorno, para poner GPU (gráfica para imágenes), p.e. Funciona con Python siempre. Y se puede copiar (guardar) en Drive, porque no se puede guardar en Colab. Por defecto, cuando se abre un nuevo archivo de notebook ya lo guarda automáticamente en Google Drive. Se puede también guardar en GitHub. O también se puede descargar como .ipynb.

También se pueden subir archivos desde tu local si le das a la carpetita.

Para abrir desde GitHub un notebook darle a archivo abrir cuaderno y copiar url de GitHub, por ejemplo.

Search ‘Stack Overflow’ como un buscador para que te ayude a corregir errores de código.

Para aprender a usar Python usar un notebook de Google, como Python Basics.

**Jupyter** es otro notebook, se abre desde Anaconda (también se puede probar desde Internet, “Jupyter Try”) y se da a launch y a New y se elige el lenguaje, que para nosotros será Python 3 (no es como Colab donde no se elige y siempre será Python).

Los snippets son ejemplos de código para hacer cosas que ya te da Colab, pero por ejemplo Jupyter no tiene las mismas librerías que Google, por lo que habrá que cargarlas primero (!pip install vega\_datasets, por ejemplo).

Por tanto, si hay error de módulo ‘x’ hay que instalar x con el !pip install x.

* Estado es explícito: solo con darle a ejecutar todo debería salirte todo bien, y si falta algo se añade para que salga.
* Escribir conclusiones porque estamos haciendo un análisis, hay que comentar un poco (no es un IDE). El notebook es para ser leído por personas, no por programadores. Nos interesa el análisis, no el código solo.
* Etiquetar los diagramas: nombrar los ejes…

Un Data Engineer usará más **Apache Zeppelin**, un Scientist usará más Jupyter o Colab. Apache tiene menos comunidad y es multi-lenguaje genuinamente (puede cada párrafo estar en uno diferente).

Para hacer ‘cd’ de algo con espacios hay que ponerlo entre comillas para que no crea que son diferentes argumentos. Aunque mejor no poner espacios como nombres de carpetas, ni caracteres extraños.

1. Clone Repo edem2021 de Pedro Nieto Git.
2. Hacer pull si no se tiene toda la info en local de ese directorio edem2021.
3. Sesiones/zeppelin/Exercise0 (no poner espacios en nombres por el puto Mac).
4. Hacer el docker run que figura en el README de edem2021 (ahí en ese directorio). Abrir primero el docker, obvio.
5. Localhost:19999, que nos lleva a Zeppelin.

Las gráficas se guardan, aunque los datos se borren (no se puede ejecutar si reinicias porque no estarán los datos). Y se pueden convertir los notebooks a pdf también.

##EJERCICIO 1: Sesión notebooks

El enunciado del ejercicio es el siguiente:

* Haced un pull de la nueva versión del repositorio ya que he corregido el notebook de Fifa
* Levantar Apache Zeppelin con el fichero de Fifa en el sitio correcto (en Ej3)
* Importar el notebook del ejercicio 3 de Fifa Analysis y visualizarlo
* Podéis pasarme una captura de pantalla aquí para ver que habéis podido