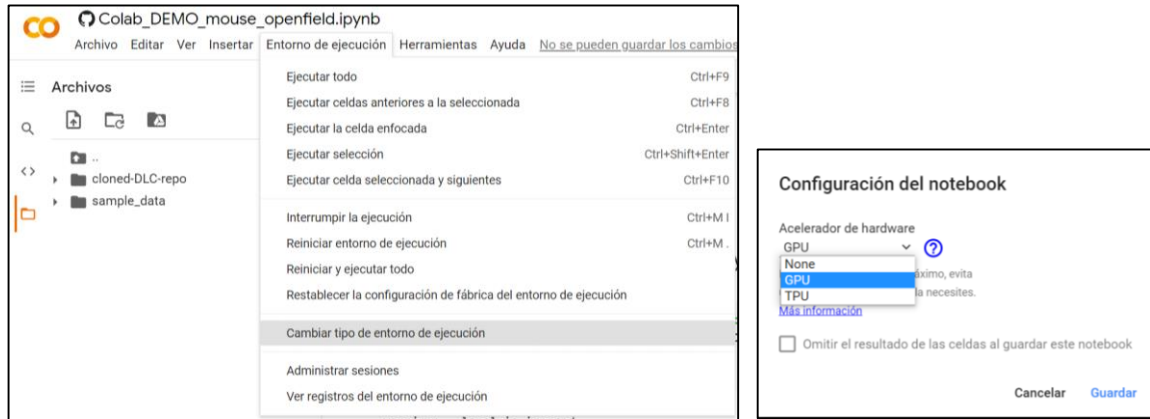


DeepLabCut en Talleres OS

ACTIVIDAD PRÁCTICA

https://colab.research.google.com/github/AlexEMG/DeepLabCut/blob/master/examples/COLAB_DEMO_mouse_openfield.ipynb

1. Lo primero que tenemos que hacer es “decirle” a Colab que vamos a trabajar con GPU.



2. Como vamos a trabajar con datos que se encuentran en la nube, en este caso en github, los dos primeros *chunks* de código nos van a pedir que clonemos estos datos para que el script los tenga disponibles cuando los necesite.

```
[ ] # Clone the entire deeplabcut repo so we can use the demo data:
!git clone -l -s git://github.com/AlexEMG/DeepLabCut.git cloned-DLC-repo
%cd cloned-DLC-repo
!ls

[ ] %cd /content/cloned-DLC-repo/examples/openfield-Pranav-2018-10-30
!ls
```

3. Luego vamos a tener que instalar DeepLabCut.

```
!pip install deeplabcut
```

Tengan en cuenta que DeepLabCut se sostiene sobre muchas librerías distintas de Python por lo que la instalación puede tomar algunos minutos.

4. Ahora es necesario que, en el output del chunk anterior clickeen “restart runtime” (esto es equivalente a restablecer la máquina virtual en la que se está ejecutando colab).
5. El próximo paso es seleccionar la versión de Tensor Flow que vamos a usar.

```
[ ] # Use TensorFlow 1.x:
%tensorflow_version 1.x
```

6. Dado que no podemos correr el GUI sobre colab, lo vamos a desactivar.

```
#GUIs don't work on the cloud, so we will suppress wxPython:
import os
os.environ["DLClight"]="True"

import deeplabcut
```

7. ¡Último paso del setting! Vamos a crear un path hacia el archivo de configuración y lo vamos a cargar.

```
[ ] #create a path variable that links to the config file:
    path_config_file = '/content/cloned-DLC-repo/examples/openfield-Pranav-2018-10-30/config.yaml'

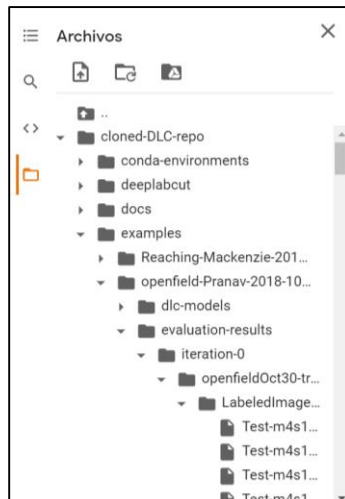
    # Loading example data set:
    deeplabcut.load_demo_data(path_config_file)
```

8. En este paso vamos a empezar a entrenar nuestra red. Cuando empiecen a correr este chunk de código van a empezar a ver que les enumera la cantidad de iteraciones y si bien lo ideal sería que corramos aproximadamente 200.000 iteraciones o mas, para los fines del taller vamos frenar en aproximadamente 1000 iteraciones.

Tal como les indica el código comentado, si ustedes no lo frenan va a seguir iterando así que vayan siguiéndolo para frenar la corrida del código. Van a ver, también, que cuando lo frenen van a obtener un error de “Keyinterrupt” no se preocupen por eso ya que no nos va a afectar.

```
deeplabcut.train_network(path_config_file, shuffle=1, displayiters=10, saveiters=100)
```

- a. Intenten buscar las imágenes con los labels y controlar si los puntos están bien ubicados.



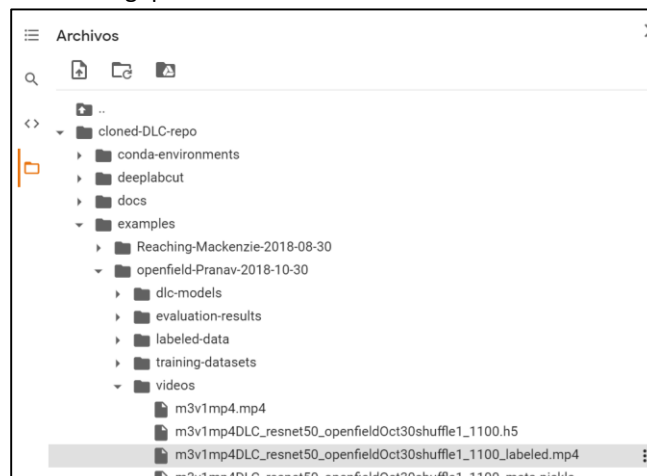
9. Una vez que tenemos los labels pasamos a analizar el video

```
videofile_path = ['/content/cloned-DLC-repo/examples/openfield-Pranav-2018-10-30/videos/m3v1mp4.mp4'] #Enter
deeplabcut.analyze_videos(path_config_file, videofile_path, videotype='.mp4')
```

10. Ahora vamos a crear un video con los labels para poder visualizarlo.

```
[ ] deeplabcut.create_labeled_video(path_config_file, videofile_path)
```

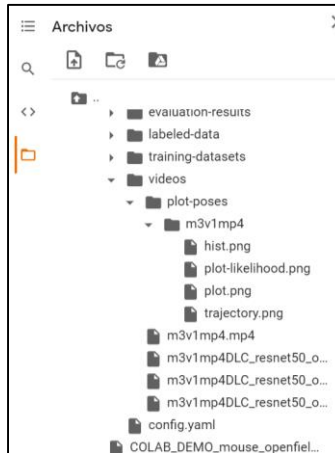
- a. Busquen el output y observen los resultados. Van a ver que los resultados no son tan buenos ya que hicimos pocas iteraciones y dependemos de un labelling que no realizamos nosotros.



11. Por último, vamos a crear gráficos para visualizar nuestros datos.

```
deeplabcut.plot_trajectories(path_config_file,videofile_path)
```

a. Busquen sus outputs.



BONUS

¿Y si probamos con otro dataset?

Dentro del mismo colab pueden seguir las instrucciones en
https://github.com/DeepLabCut/DeepLabCut/blob/master/examples/Demo_labeledexample_MouseReaching.ipynb