



# **Guía Completa para la Creación de un Dataset de Detección de Objetos con Roboflow y entrenamiento de un modelo.**

## **Introducción a la Detección de Objetos y Roboflow**

La detección de objetos es una tecnología de visión por computadora que permite identificar y localizar objetos específicos dentro de imágenes o videos. A diferencia de la simple clasificación de imágenes, la detección de objetos no solo determina qué objetos están presentes, sino también dónde se encuentran mediante la generación de cuadros delimitadores (bounding boxes) o en nuestro caso segmentaciones.

Roboflow es una plataforma integral que facilita todo el proceso de creación de modelos de detección de objetos, desde la recopilación y anotación de datos hasta el entrenamiento e implementación de modelos. Es especialmente útil para trabajar con modelos YOLO (You Only Look Once).

En esta guía, explicaremos cómo utilizar Roboflow para crear un dataset de detección de objetos desde cero, anotar imágenes y entrenar un modelo YOLO para utilizarlo posteriormente en nuestro software desarrollado.

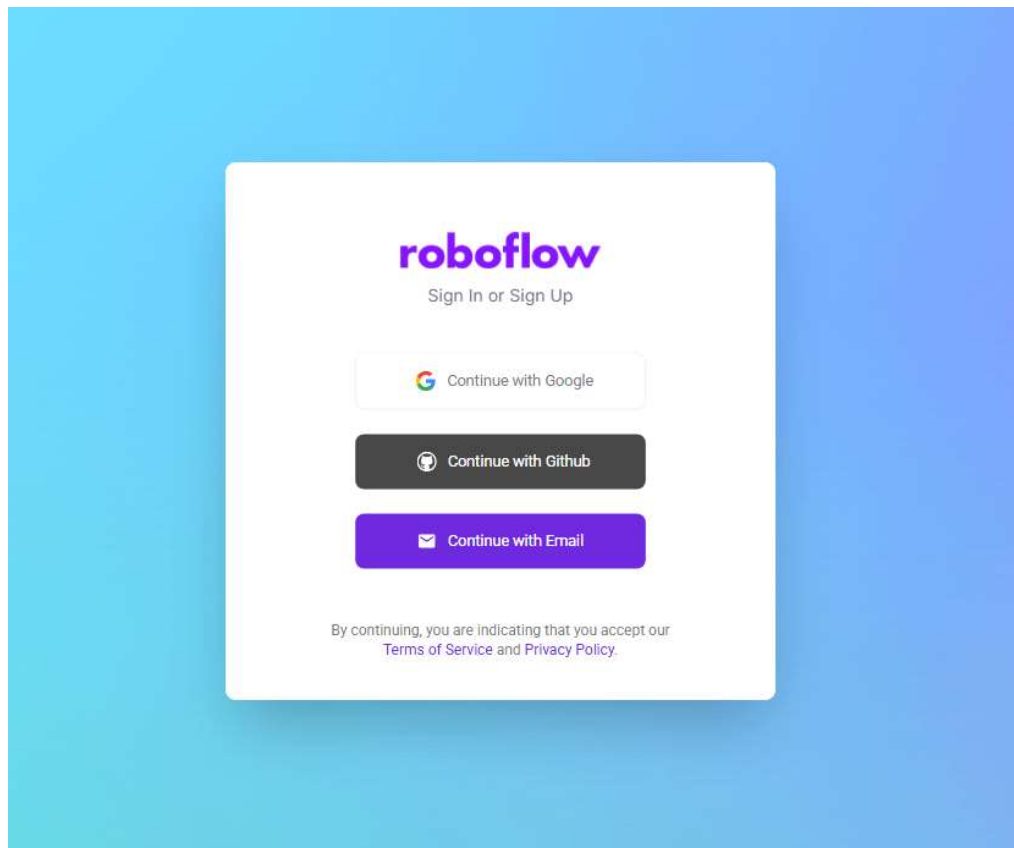
***Toda la información que se encuentre en esta guía puede ser complementada con la documentación que ofrece Roboflow (<https://docs.roboflow.com/>).***

## **1. Configuración Inicial en Roboflow**

### **1.1 Creación de una Cuenta en Roboflow**

Para comenzar a utilizar Roboflow, es necesario crear una cuenta:

- Visita [roboflow.com](https://roboflow.com)
- Haz clic en "Sign Up" o "Get Started" en la esquina superior derecha.
- Puedes registrarte con tu correo electrónico o mediante integración con Google, GitHub o LinkedIn



- Completa el proceso de registro proporcionando la información requerida
- Verifica tu correo electrónico si es necesario

La versión gratuita de Roboflow ofrece características suficientes para proyectos pequeños o de prueba, pero existen planes de pago con capacidades adicionales para proyectos más grandes o equipos.

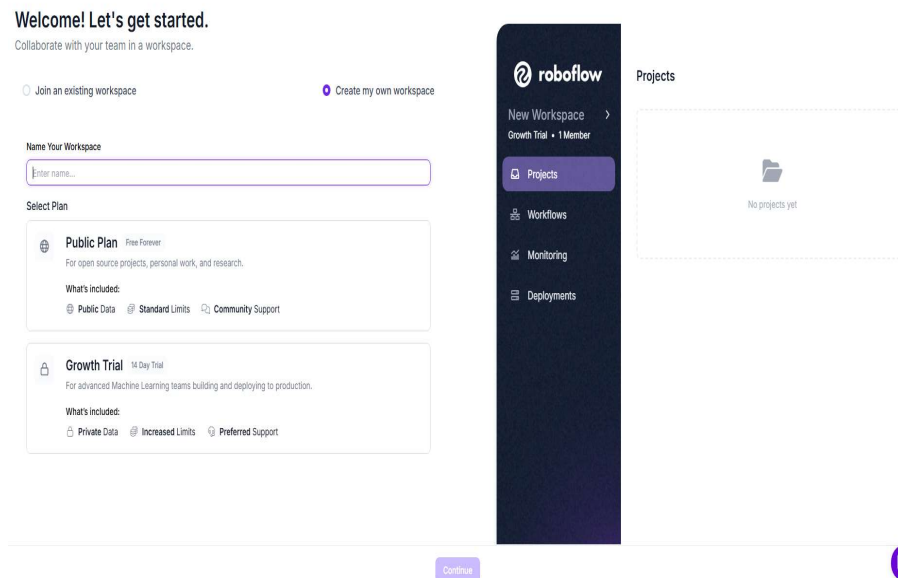
	MOST POPULAR	
<b>Basic Plan</b> For small teams <b>\$65</b> per month, billed monthly <input type="button" value="Upgrade to Basic Plan"/> Data and Models are <b>private</b> . Everything in Public Plan, plus: <ul style="list-style-type: none"> <li>✓ 30 credits per month, loaded upfront</li> <li>✓ 5 User Seats</li> <li>✓ Pay As You Go Training and Storage</li> <li>✓ Inference Model License</li> <li>✓ Model Weights Download</li> <li>✓ Model Evaluation</li> <li>✓ Community Support</li> </ul>	<b>Growth Plan</b> For startups <b>\$399</b> per month, billed monthly <input type="button" value="Upgrade to Growth Plan"/> Data and Models are <b>private</b> . Everything in Basic Plan, plus: <ul style="list-style-type: none"> <li>✓ 150 credits per month, loaded upfront</li> <li>✓ 20 User Seats</li> <li>✓ Role-Based Access Control</li> <li>✓ Professional Labeling Services</li> <li>✓ Premium GPU Access</li> <li>✓ Downloaded Model License</li> <li>✓ Priority Chat and Email Support, Onboarding with Specialists</li> </ul>	<b>Enterprise Plan</b> For <b>Custom Pricing</b> Billed annually <input type="button" value="Contact Sales"/> Data and Models are <b>private</b> . Everything in Growth Plan, plus: <ul style="list-style-type: none"> <li>✓ Custom user limits with Single Sign-On</li> <li>✓ Model Monitoring</li> <li>✓ Device Management</li> <li>✓ Hardware Procurement</li> <li>✓ Offline Deployments</li> <li>✓ SSO + Scoped API Keys</li> <li>✓ Priority Access to Premium GPL</li> <li>✓ HIPAA Compliance + BAA Execution</li> <li>✓ Dedicated Field Engineering Support</li> </ul>

Existen dos versiones pagas de Roboflow, Basic Plan, de 65u\$d por mes y Growth Plan de 399u\$d por mes. En nuestro caso podremos utilizarlo de manera gratuita con limitaciones, que no van a ser un gran impedimento para la creación

del modelo.

## 1.2 Creación de un Nuevo Proyecto

La primera vez que iniciemos sesión en Roboflow veremos el siguiente panel:



En donde crearemos un Workspace o podremos unirnos a un Workspace existente, en el caso que seamos invitados a un Workspace nos aparecerá la invitación dentro de esta lista.

En mi caso crearé un Workspace bajo el plan Public Plan (que es el plan gratuito que ofrece Roboflow) con el nombre “*hojas-guia*”.

Una vez creado el workspace tendremos la oportunidad de invitar a 2 personas mas que se sumarán al proyecto y podrán anotar, subir o descargar imágenes y desplegar modelos, dependiendo el rol que se le asigne en la invitación.

## Invite teammates.

Add collaborators to help with labeling, upload data, train models, and more.

✉ 2 invites available

email@example.com

email@example.com

email@example.com

Los roles disponibles son:

- Admin: Tendrá control total sobre el workspace
- Labeler: Será el rol que tendrán los anotadores.
- Reviewer: Tendrá mayor control sobre las anotaciones, pudiendo aprobar o desaprobadas, reasignar lotes de anotación a los labelers, etc.

Una vez que hayas iniciado sesión en Roboflow, veremos el panel:



- En el panel de control, haz clic en "+ New Project"
- Le asignaremos un nombre a nuestro proyecto y el nombre del grupo de trabajo.

## Let's create your project.

hojasguia > New Public Project

Project Name

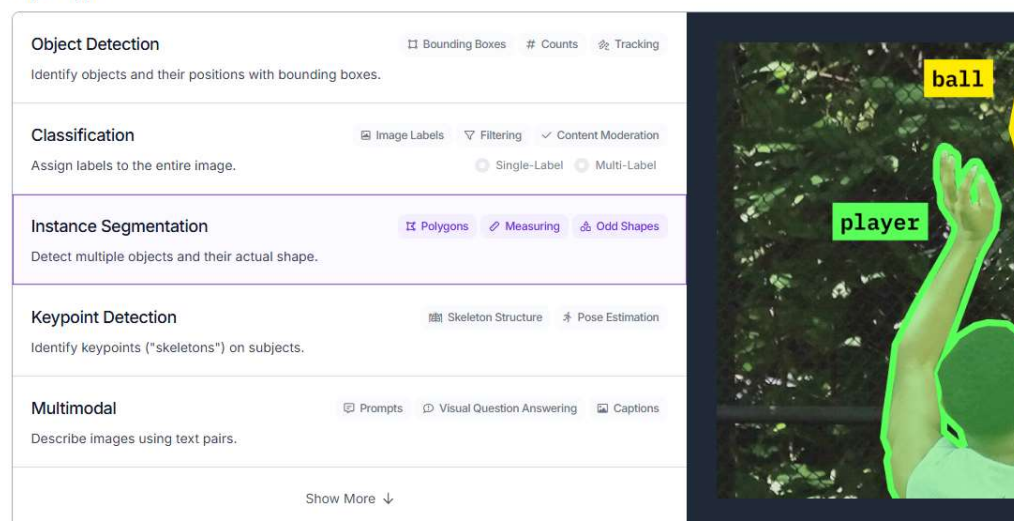
GuiaRoboflow

Annotation Group

hojas

- Selecciona el tipo de proyecto como "Instance Segmentation", lo que nos permitirá anotar con polígonos los objetos que nos interesan en las imágenes.

Project Type



- Selecciona una licencia para tu dataset (en este caso deberá ser una licencia opensource ya que el plan gratuito de Roboflow no permite aplicar licencias privadas. En nuestro caso dejamos CC BY 4.0)
- Para finalizar le damos clic en "Create Public Project"

### 1.3 Configuración del Proyecto

Después de crear el proyecto, es importante definir las clases de objetos que deseas detectar:

- Ve a la sección "Classes & Tags" en el menú lateral izquierdo.
- En el cuadro de texto podremos definir la clase del objeto que el modelo va a detectar (En nuestro proyecto recomendamos utilizar el nombre de clase "hojas").

☰ Classes & Tags

Add a comma separated list of class names

hojas

☐ Fix invalid classes

- Una vez que demos click en "Add Classes" podremos asignarle un color distintivo a cada clase para facilitar la visualización durante la anotación. Como en este caso serán anotaciones de una sola clase es indiferente el color elegido.

## 2. Recopilación y Carga de Imágenes

### 2.1 Consideraciones para un Dataset Efectivo

Antes de comenzar a recopilar imágenes, es importante considerar algunos principios para crear un dataset efectivo:

- **Diversidad:** Incluye imágenes en diferentes condiciones (iluminación, ángulos, etc.)
- **Representatividad:** Las imágenes deben representar los escenarios donde se utilizará el modelo
- **Cantidad:** Para modelos iniciales, se recomienda al menos 50-100 imágenes por clase; para modelos robustos, 500+ por clase
- **Calidad:** Imágenes nítidas y con resolución adecuada (no excesivamente grandes ni pequeñas)

### 2.2 Carga de Imágenes en Roboflow

Roboflow ofrece varias opciones para cargar imágenes:

- **(Caso 1) Carga directa:**
  - En tu proyecto, en el panel izquierdo, selecciona "Upload Data"
  - Arrastra y suelta imágenes o selecciona "Select Files"

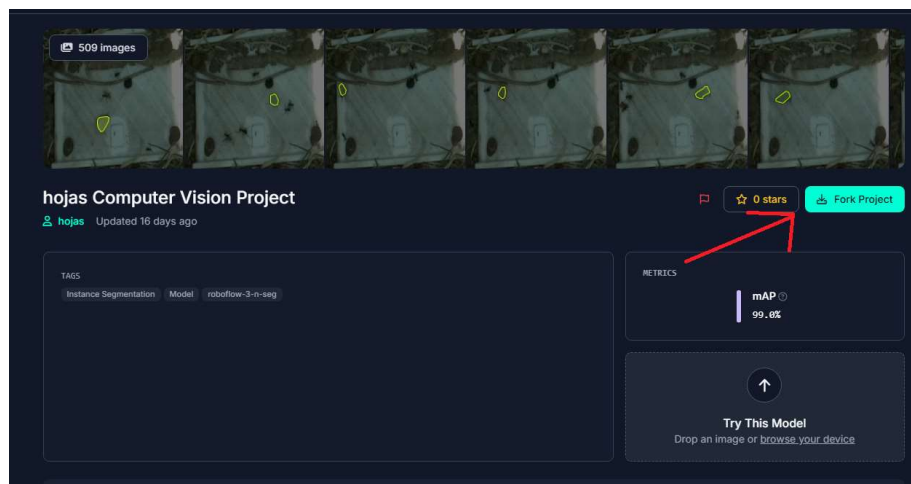
- Presionamos “Save and Continue”.
- Espera a que se complete la carga.

Esto cargará imágenes a nuestro proyecto y luego aparecerán bajo la categoría “Unassigned” para comenzar la anotación, paso que explicaremos mas adelante en la guía (Sección 3).

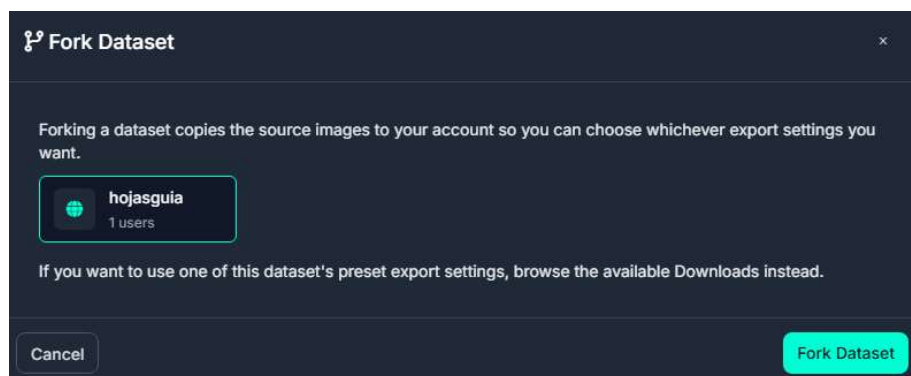
- **(Caso 2) Importación desde otro dataset:**

En nuestro caso ya creamos un dataset, teniendo esto en cuenta, esta es la opción más recomendada para comenzar a trabajar en Roboflow. Para obtener el dataset “hojas” (<https://universe.roboflow.com/hojas/hojas-tbrad>) dentro de nuestro proyecto deberemos:

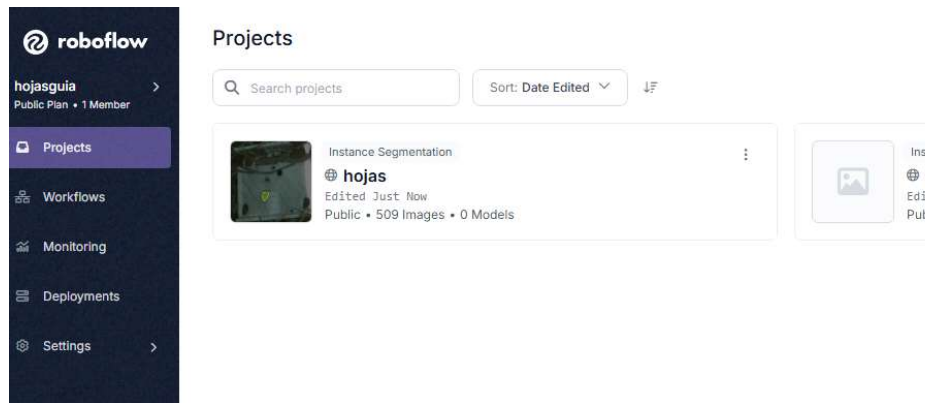
- Ingresar al link <https://universe.roboflow.com/hojas/hojas-tbrad>
- Le daremos click a “Fork Project”, esto nos permitirá copiar las anotaciones a nuestro Workspace y podremos acceder a cada una de ellas.



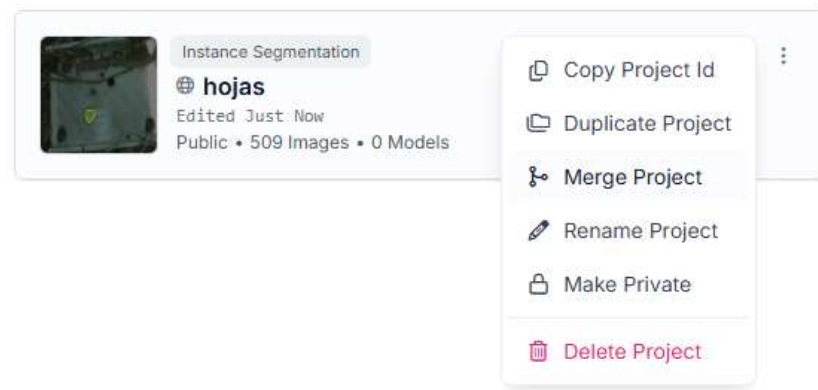
- Elegimos el workspace de destino y le damos click a “Fork Dataset”.



- Una vez de vuelta a nuestro Workspace, le daremos click en el panel izquierdo en “Projects” y veremos dos proyectos, el creado anteriormente y la copia del dataset.



- Ahora uniremos ambos proyectos en uno solo, dándole click a los tres puntos verticales que aparecen en los proyectos y seleccionando “Merge Project”



- Lo cual nos permitira seleccionar nuestros dos proyectos y crear uno nuevo con los datos de ambos.
- Elegiremos un nombre:



Merge Projects

New Project Name

Project Type

Instance Segmentation

New Annotation Group ?

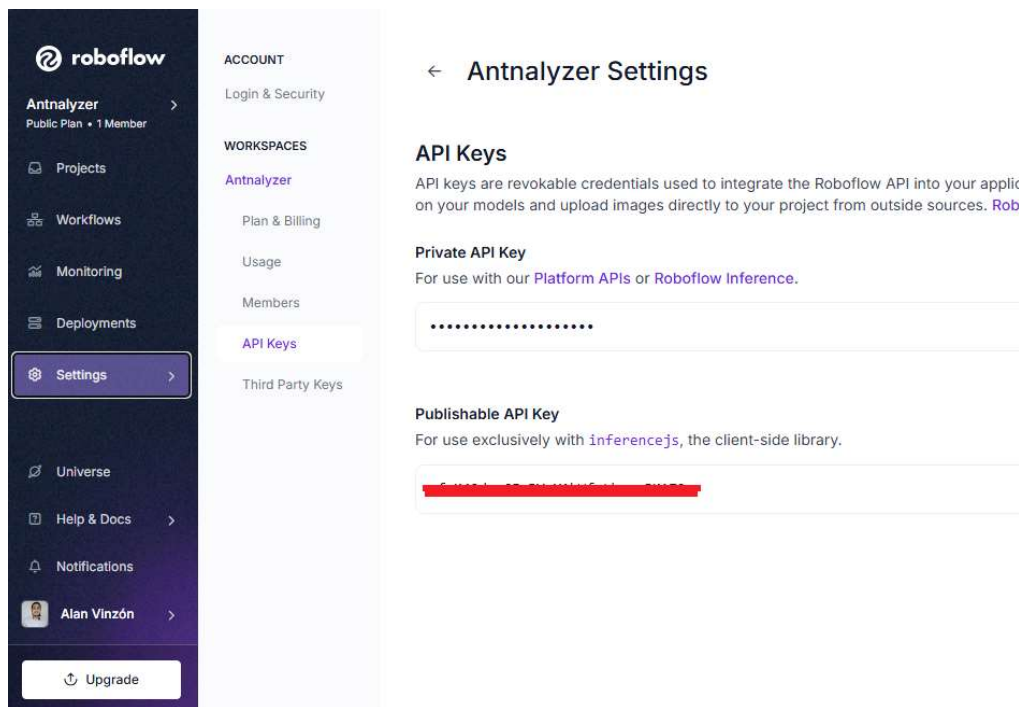
hojas

Cancel Merge Projects

***Cabe aclarar que al clonar un dataset existente no es necesario crear desde cero un nuevo proyecto anteriormente, sino que desde un comienzo puede usarse el dataset clonado desde nuestro panel de Workspace.***

- **(Caso 3) Mediante API:**

Podremos utilizar la API Key de nuestra cuenta para acceder a nuestros proyectos y realizar carga y descarga de imágenes o anotaciones. Para visualizar nuestra API en el panel izquierdo le daremos click a “Settings” y luego “API Keys”. La que nos interesa es la “Private API Key” la cual servirá como llave para poder ejecutar funciones de roboflow en nuestra cuenta de forma remota a través de código.



La API Key es un elemento privado que se debe tener mucho cuidado al compartir y no debe ser transferido a personas que no son de nuestra confianza. Por ejemplo, el código (python) que puede ser implementado para subir imágenes desde nuestra API Key será

```
from roboflow import Roboflow
rf = Roboflow(api_key="tu_api_key")
project = rf.workspace("tu_workspace").project("nombre_proyecto")

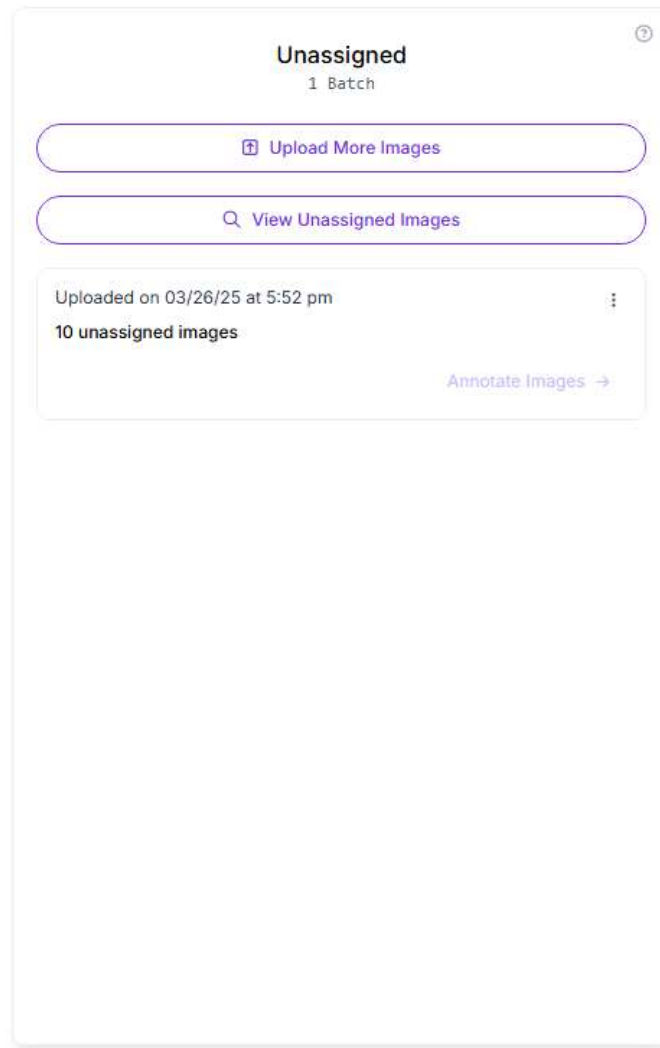
# Cargar una imagen
project.upload("ruta/a/imagen.jpg")

# Cargar múltiples imágenes
import os
for imagen in os.listdir("carpeta_de_imagenes"):
    if imagen.endswith((".jpg", ".jpeg", ".png")):
        project.upload(f"carpeta_de_imagenes/{imagen}")
```

### 2.3 Organización del Dataset

Una vez cargadas las imágenes, Roboflow las coloca por defecto en el conjunto "Unassigned".

Cargaremos 10 imágenes para utilizar como ejemplo. Como podemos ver en la imagen tenemos 10 imágenes sin asignar



### 3. Anotación de Imágenes

La anotación es el proceso de marcar los objetos en las imágenes para que el modelo aprenda a reconocerlos. Roboflow ofrece herramientas integradas para este propósito.

#### 3.1 Acceso al Editor de Anotaciones

Para comenzar la anotación:

- En el panel principal del proyecto, haz clic en "Annotate"
- Se mostrará la lista de imágenes sin asignar disponibles. Estas imágenes deben ser pasadas al proceso de anotación.
- Damos click al lote de imágenes sin asignar (Unassigned) y en la esquina superior derecha damos click a "Annotate Images"
- De un menú deberemos seleccionar entre 3 opciones:

## How do you want to label your images? ×

### Auto Label Beta

Use your custom model or a zero-shot model to label an entire batch.

 Start Auto Label

### Manual Labeling

You and your team label your own images with help from our AI labeling tools.

 Start Manual Labeling

### Roboflow Labeling Service

Work with a professional team of human labelers.

 Get Details

- Autolabel: Selección automatizada con un modelo pre-entrenado.
- Manual Labeling: anotación manual por el usuario
- Roboflow Labeling: Roboflow ofrece un grupo de personas para que anoten en el lugar del usuario.

**Manual Labeling** será la herramienta que tendremos que utilizar para segmentar las imágenes.

- Una vez seleccionado Manual Labeling tendremos la opción de asignar a integrantes de nuestro grupo imágenes a etiquetar o podemos asignarlas a nosotros mismos.
- Seleccionamos la primer imagen de la lista y comenzamos con el anotado

### 3.2 Creación de Anotaciones Manuales

En el proceso de anotación manual tendremos varias herramientas a nuestra disposición, que se encuentran en el panel derecho, tal como se observa en la imagen:



Las herramientas son:

- Drag (Atajo D): Nos permitira mover la imagen dentro del espacio de trabajo
- Bounding Box (Atajo B): será utilizada para marcar con cajas delimitadoras los objetos. En nuestro caso es un modelo de segmentación entonces no usaremos esta herramienta.
- Polygon Tool (Atajo P): Con esta herramienta podremos dibujar anotaciones en forma libre y precisa. Esta es la principal herramienta que utilizaremos para realizar las anotaciones.
- Smart Polygon (AI Labeling)(Atajo S): Es una herramienta de Roboflow que asiste en la creación de la anotación. Para objetos uniformes es una herramienta que facilita la creación de los poligonos.
- Label Assistant (AI Labeling): Nos permitira utilizar un modelo pre-entrenado para detectar los elementos en nuestras imágenes.
- Repeat Previous (Atajo R): Repetirá las anotaciones previas.
- Herramienta de comentarios (Atajo C).
- Deshacer.
- Rehacer.
- Marcar imagen como nula (Atajo N).

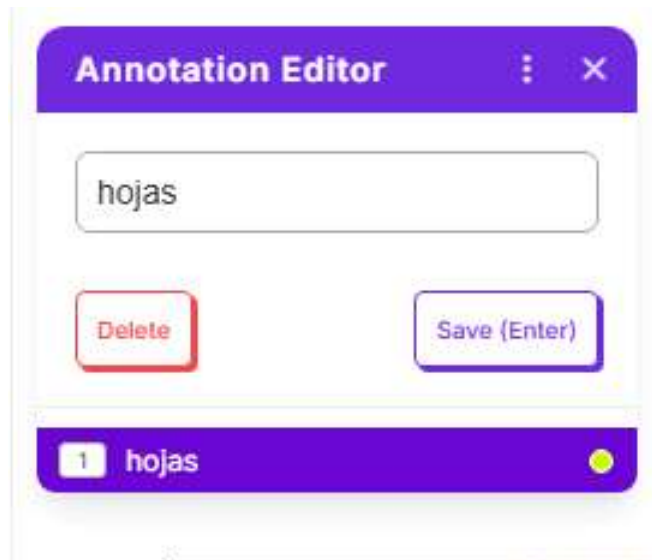
En la parte izquierda veremos las clases disponibles para la anotación. Con

la ayuda de las utilidades que nos provee Roboflow ya podremos comenzar con la anotaciones y buscar la herramienta que mejor se adecue a nuestras necesidades.

Nos deberá quedar el objeto que buscamos detectar encerrado en un polígono, tal como se observa en la imagen:



Luego se debe presionar “Save (Enter)”



y en la parte superior podremos cambiar a la siguiente imagen a etiquetar:



Para agilizar el proceso de anotación:

- **Atajos de teclado:**
  - 1, 2, 3... para seleccionar clases
  - Ctrl+Z (Cmd+Z en Mac) para deshacer
  - Delete para eliminar una anotación seleccionada
  - A y D para navegar entre imágenes
- **Recomendaciones:**
  - Ajusta la opacidad de las anotaciones para mayor visibilidad
  - Utiliza zoom para anotar objetos pequeños con precisión

Una vez finalizadas las anotaciones podemos volver hacia atrás y observar las miniaturas de las imágenes etiquetadas, donde podremos realizar un chequeo rápido de las anotaciones para verificar que este todo en condiciones para crear el dataset.

Unannotated 0 Annotated 10



## 4. Preparación del Dataset

Una vez que hayas anotado tus imágenes, en la esquina superior derecha, dando click Add Images to Dataset agrega las imágenes al dataset.

### 4.1 Distribución en Conjuntos

Roboflow facilita la división del dataset en conjuntos de entrenamiento, validación y prueba:

- En "Dataset Split", configura los porcentajes para cada conjunto (una distribución típica es 70-20-10 o 80-10-10, para train, val y test respectivamente), como se observa en la imagen.



## Add Images To Dataset



 Total Images to Add: 10

Method

 What's Train, Valid, Test?

Split Images Between Train/Valid/Test



Train  
70%

Valid  
20%

Test  
10%



Image Distribution

Train: 7 images

Valid: 2 images

Test: 1 images

You are about to add **10 images** to the dataset.  
**0 images** will be sent back as part of a new job.

- También puedes asignar manualmente imágenes específicas a cada conjunto

### 4.2 Preprocesamiento de Imágenes

Una vez hayamos pasado todas nuestras anotaciones al dataset, deberemos crear una nueva versión del mismo. Accederemos al panel de dataset dando click en "Dataset" en la parte izquierda del workspace. Luego en la esquina superior derecha tendremos el botón "New Dataset Version". Roboflow ofrece opciones para preprocesar las imágenes antes del entrenamiento, podremos seleccionar estas opciones en la parte de "Preprocessing" en la creación de la nueva versión.:

- **Redimensionamiento:**
  - Establece un tamaño uniforme para todas las imágenes
  - Para YOLO, los tamaños comunes son 416×416, 512×512 o 640×640 píxeles.
  - En nuestro caso es recomendable utilizar 640x640.
- **Normalización:**
  - Permite ajustar el contraste y brillo

- Escala los valores de píxeles (por ejemplo, a un rango de 0-1)

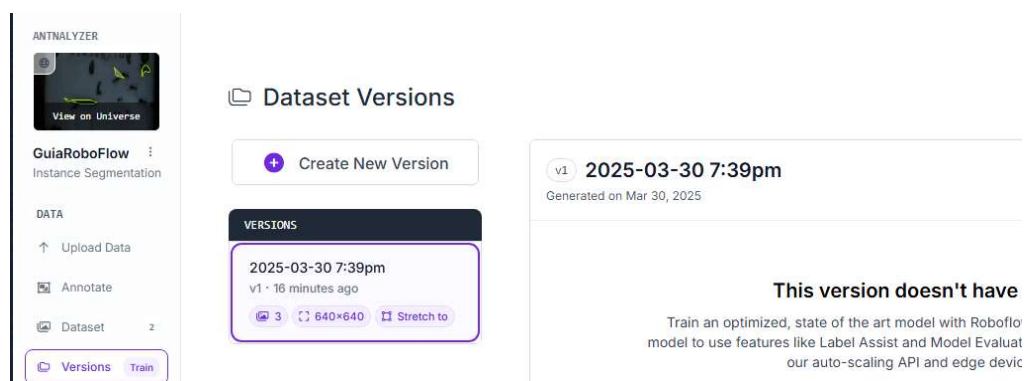
#### 4.3 Aumento de Datos

El aumento de datos es una técnica poderosa para incrementar artificialmente el tamaño del dataset y mejorar el modelo, podremos seleccionar estas opciones en la parte de "Augmentation" en la creación de la nueva versión:

- Roboflow ofrece múltiples opciones de aumento de datos:
  - **Geométricas:** Rotación, volteo, recorte, zoom
  - **Fotométricas:** Brillo, contraste, ruido, desenfoque
  - **Climáticas:** Simulación de niebla, lluvia, nieve
  - **Avanzadas:** Mosaic, CutMix, etc.
- Configura cada aumento según tus necesidades:
  - Establece los rangos o probabilidades para cada transformación
  - Previsualiza el efecto en imágenes de muestra
  - Considera qué aumentaciones son relevantes para tu caso de uso.
- Establece la cantidad de imágenes aumentadas que desees generar por imagen original. Podremos seleccionar x2 o x3 en caso de tener la cuenta gratuita.
- Una vez configurados todos los parámetros, haz clic en "Create"
- Roboflow procesará todas las imágenes aplicando el preprocesamiento y aumentaciones configuradas. Este proceso puede tomar tiempo dependiendo del tamaño del dataset y las transformaciones aplicadas
- Una vez completado, podrás revisar el dataset generado.

#### 4.4 Exportación del Dataset

En caso de querer entrenar un modelo YOLO de forma local (Sección 5.2) necesitarás exportar el dataset en el formato adecuado:



- En "Versions", selecciona la versión correspondiente y da click en "Download Dataset"

- Elegiremos el formato de la versión YOLOv8
- Se puede descargar a través de código, o en formato zip.

## 5. Entrenamiento

A partir de este momento explicaremos diferentes formas de entrenar un modelo. Existen varias formas de realizar el entrenamiento, pero para esta guía explicaremos tres, de los cuales podremos elegir una que sea la forma más conveniente.

### 5.1 (Caso 1) Entrenamiento en la nube con Google Colab

El entrenamiento de modelos con Google Colab es sencillo, fácil de aplicar y muy veloz, con la desventaja de que este tipo de procesamientos lleva un tiempo considerable, por lo cual deberemos abonar unidades computacionales o pagar una suscripción mensual en Google Colab para poder entrenar modelos sin problemas.

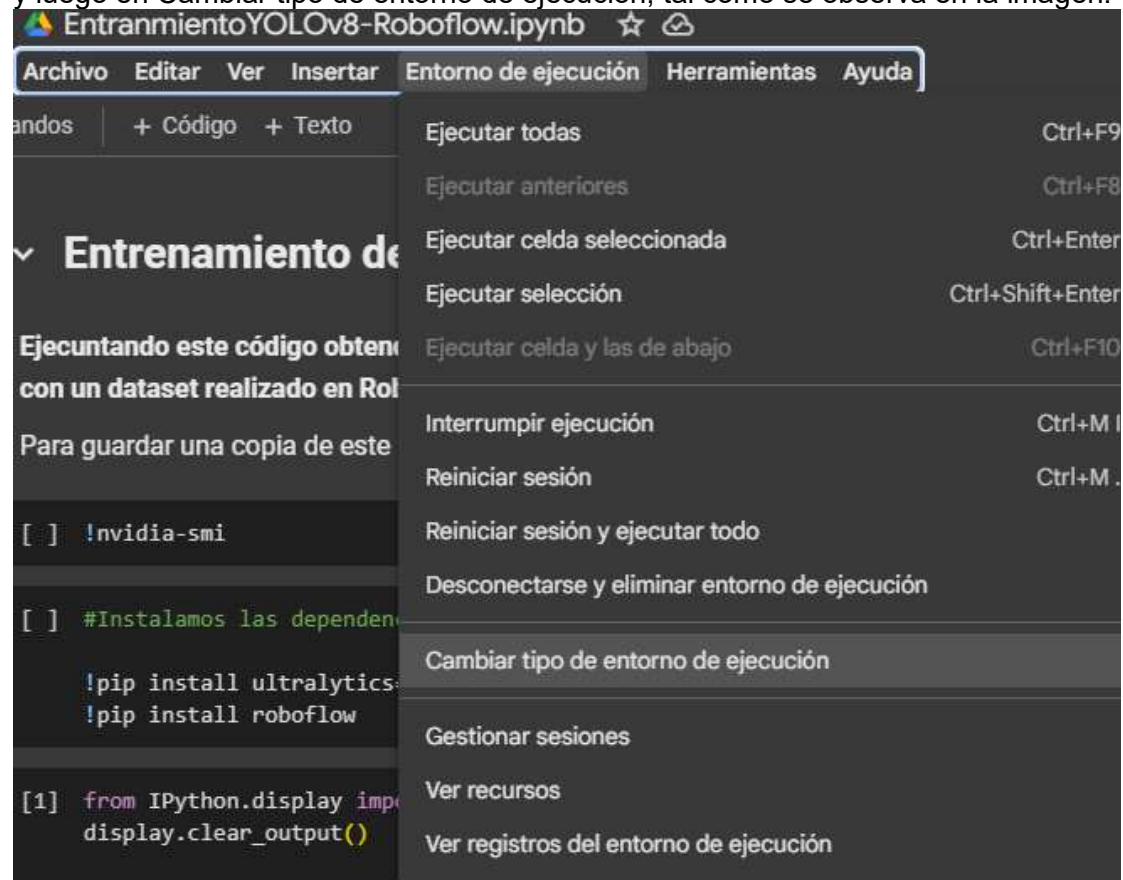
The image shows a comparison of Google Colab pricing plans. The 'Colab Pro' plan is highlighted with an orange border. The plans are as follows:

Plan	Precio	Beneficios
Pay As You Go	100 unidades de procesamiento por US\$ 9,99 500 unidades de procesamiento por US\$ 49,99	Actualmente, tienes 0 unidades de procesamiento. Las unidades de procesamiento vencen después de 90 días. Compra más cuando las necesites. ✓ No se requiere suscripción. Paga solo por lo que usas. ✓ GPU más rápidas. Actualiza a GPU más potentes.
Colab Pro (Recomendado)	US\$ 9,99 al mes	✓ 100 unidades de procesamiento por mes. Las unidades de procesamiento vencen después de 90 días. Compra más cuando las necesites. ✓ GPU más rápidas. Actualiza a GPU más potentes. ✓ Más memoria. Accede a nuestras máquinas de mayor capacidad de memoria. ✓ Terminal. Capacidad de usar una terminal con la VM conectada.
Colab Pro+	US\$ 49,99 al mes	Todos los beneficios de Pro, además de lo siguiente: ✓ Un adicional de 400 unidades de procesamiento por un total de 500 por mes. Las unidades de procesamiento vencen después de 90 días. Compra más cuando las necesites. ✓ GPU más rápidas. Acceso prioritario para actualizar a GPU premium más potentes. ✓ Ejecución en segundo plano. Si tienes unidades de procesamiento, tu notebook activo seguirá funcionando durante un máximo de 24 horas, incluso si cierras el navegador.

Evaluando el rango de precios que tiene Roboflow con los de Google Colab es quizás más conveniente abonar una suscripción en Google Colab y utilizar Roboflow de manera gratuita para solamente crear y distribuir datasets. Además tendremos la opción de pagar 100 unidades de procesamiento iniciales las cuales duran 90 días para probar el servicio. Las unidades computacionales se irán consumiendo a medida que se mantiene el entorno de Colab corriendo, es decir, si entrenamos un modelo por 2 horas puede consumir hasta 6 unidades computacionales.

Para el entrenamiento en Google Colab podemos ingresar a este link: [EntrenamientoYOLOv8-Roboflow.ipynb - Colab](#) donde está el código completo necesario para crear un modelo YOLOv8 con un dataset realizado en Roboflow. Es recomendable realizar una copia del documento para poder editarlo (Archivo > Guardar una copia en Drive).

Es importante utilizar una GPU disponible para la tarea de entrenamiento, esto lo podremos cambiar seleccionando Entorno de Ejecución en la barra de tareas y luego en Cambiar tipo de entorno de ejecución, tal como se observa en la imagen.



Luego veremos un panel donde veremos las GPUs disponibles para nuestro entorno de ejecución, en caso de haber pagado una suscripción habrá disponibilidad de GPUs con mayor rendimiento.

## Cambiar tipo de entorno de ejecución

Tipo de entorno de ejecución

Python 3 ▼

Acelerador por hardware ?

☐ CPU
 ☒ GPU T4
 ☐ GPU A100
 ☐ GPU L4

☐ TPU v2-8
 ☐ TPU v5e-1

¿Quieres acceder a GPUs premium?

[Compra unidades de computación adicionales](#)

Cancelar
 Guardar

En mi caso al no poseer una suscripción ni unidades de computación adicionales solamente puedo acceder a una GPU T4.

Ejecutamos los pasos en orden:

```
!nvidia-smi
```

```
Sun Mar 30 22:49:57 2025
```

NVIDIA-SMI 550.54.15		Driver Version: 550.54.15		CUDA Version: 12.4	
GPU	Name	Persistence-M	Bus-Id	Disp.A	Volatile Uncorr. ECC
Fan	Temp	Pwr:Usage/Cap	Memory-Usage	GPU-Util	Compute M. MIG M.
0	Tesla T4	Off	00000000:00:04.0	Off	0
N/A	43C	9W / 70W	0MiB / 15360MiB	0%	Default N/A

```

+-----+
| Processes: |
| GPU   GI   CI        PID   Type   Process name                      GPU Memory |
| ID   ID   ID           |          |                         Usage      |
+-----+
| No running processes found |
+-----+

[ ] #Instalamos las dependencias

!pip install ultralytics==8.2.103 -q
!pip install roboflow

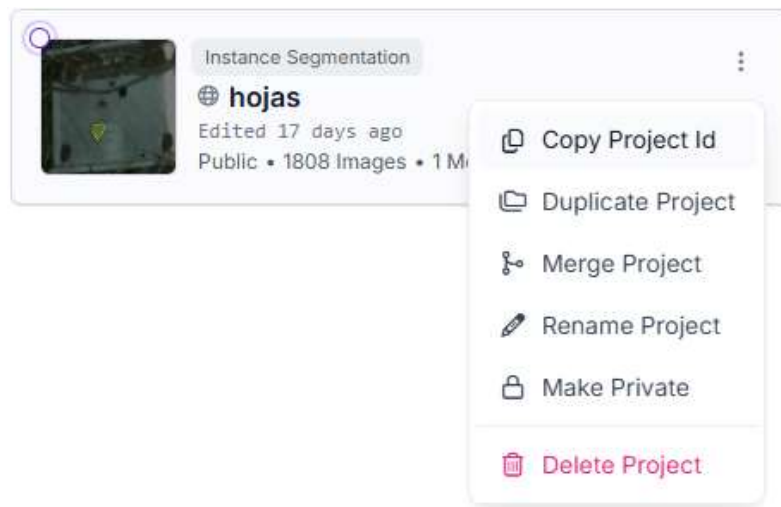
```

Al llegar a la seccion:

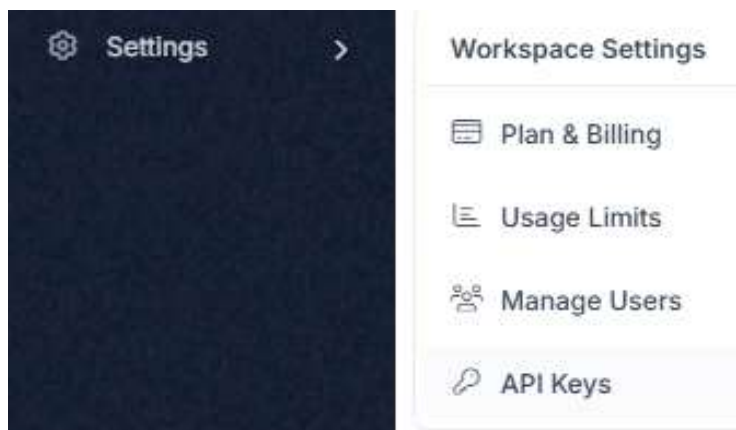
```
!mkdir -p {HOME}/datasets
%cd {HOME}/datasets

rf = Roboflow(api_key="API Key Propia")
project = rf.workspace("hojas").project("hojas-tbrad") #Nombre del workspace y nombre del proyecto en nuestro caso "hoja
version = project.version(4) #Version del proyecto
dataset = version.download("yolov8") #Descargaremos el dataset para entrenamient con YOLOv8
```

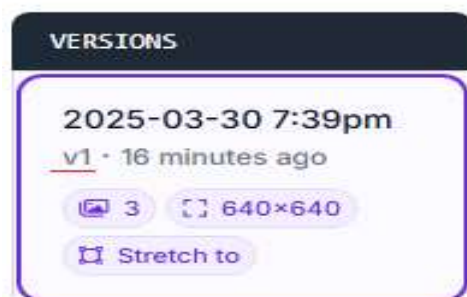
Cambiando los valores de API Key y el nombre y versión del proyecto podremos entrenar nuestro modelo ejecutando cada celda de código. El nombre del proyecto lo podremos obtener dando click en “Copy Project Id” como se observa en la imagen.



La Api Key la obtendremos de la seccion Settings>Api Keys y utilizaremos la Private API KEY

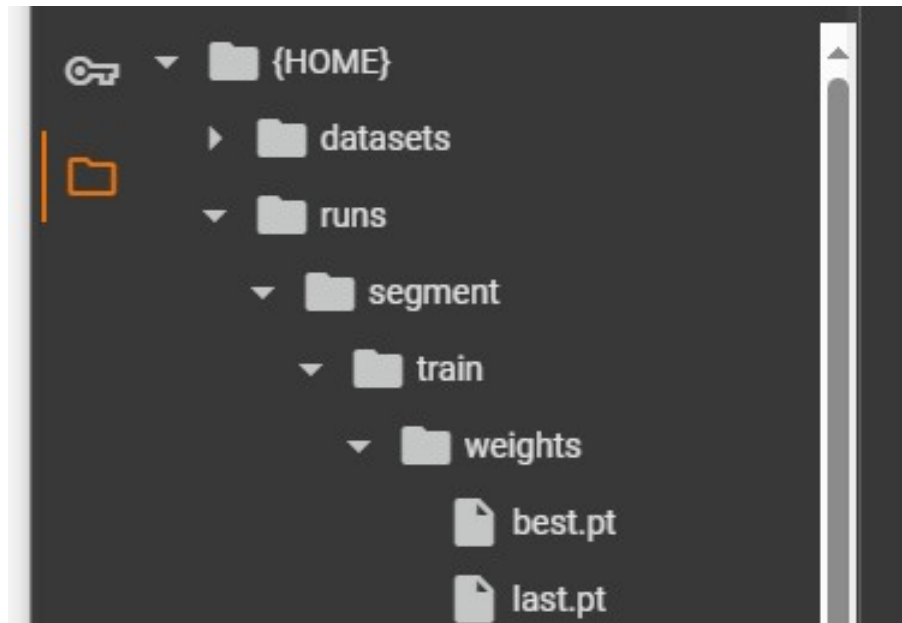


y la version la determinaremos segun el dataset version que corramos (en este caso, 1)



Debemos tener en cuenta que, una vez entrenado el modelo deberemos

descargar los archivos resultantes del entrenamiento a nuestra PC o moverlo a una carpeta en drive para no perderlos, que se encontrará en el directorio `/runs/segment/train` dentro de Colab, tal como se observa en la imagen.



Dentro de la carpeta `train` estará la carpeta `weights` donde habrá dos archivos, un archivo denominado `"best.pt"` que será el modelo con mejores métricas y `"last.pt"` que será el último modelo entrenado.

Luego de esto estamos listos para utilizarlo.

## 5.2 (Caso 2) Entrenamiento Local con Código Generado

Si es preferible entrenar el modelo en tu propia computadora deberemos tener en cuenta que el entrenamiento local requiere una GPU con suficiente memoria para un rápido entrenamiento y resultados eficientes. También se necesitan conocimientos sobre la consola de windows, ya que desde ahí instalaremos dependencias y ejecutaremos los comandos para el entrenamiento.

La ventaja de este tipo de entrenamiento reside en que no existe la necesidad de un pago o una suscripción y también no se necesita una conexión a internet, pero estaremos utilizando los recursos de nuestra PC lo cual puede ser perjudicial para la vida útil del equipo, como así también, el uso de un alto porcentaje de la GPU puede provocar que la PC no pueda ser utilizada durante un entrenamiento.

- En la sección de exportación, selecciona "Download Zip". Esto nos permite descargar las carpetas de `train`, `val` y `test`.
- Debemos comprobar que el archivo `data.yaml` tenga el directorio de las carpetas donde se encuentran `train` y `val`. Es una buena práctica escribir el directorio completo.
- Se necesitarán instalar las dependencias necesarias:
  - YOLOv8: `pip install ultralytics==8.2.103 -q` (es importante mantener la versión de `ultralytics` para que coincida con la versión utilizada en el



software)

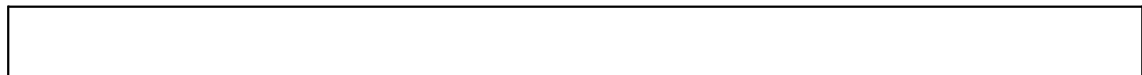
- Roboflow: pip install roboflow
- Roboflow proporciona un código de ejemplo para entrenar localmente
- Para YOLOv8, el código será:

```
from ultralytics import YOLO

# Cargar un modelo (new=True significa crear un modelo nuevo)
model = YOLO('yolov8n.pt') # carga un modelo pre entrenado
                             (recomendado para transfer learning)

# Entrenar el modelo con tus datos
results = model.train(
    data='ruta/a/tu/data.yaml', # archivo de configuración del
    dataset                      # dataset
    epochs=100,                  # número de epochs
    imgsz=640,                   # tamaño de las imágenes
    batch=16,                    # tamaño del batch
    name='mi_modelo'             # nombre del modelo
)
```

- También puede utilizarse la API de Roboflow para crear un programa que descargue el dataset y luego comenzar el entrenamiento desde la consola.



Este código descarga el dataset en nuestra PC, luego abriremos la consola, con la ayuda del comando “pushd” nos posicionamos en el directorio donde está el dataset y luego ejecutamos el comando

```
yolo task=segment mode=train model=yolov8m-seg.pt data=data.yaml
epochs=100 imgsz=640 batch=12
```

Donde:

- epochs: Número de veces que el algoritmo revisará todo el conjunto de datos.
- imgsz: Tamaño al que se redimensiona las imágenes durante el entrenamiento (640×640 píxeles). Este valor equilibra la captura de detalles con la eficiencia computacional.
- batch: Cantidad de imágenes que se procesan simultáneamente antes de actualizar los pesos del modelo. Si tenemos una buena CPU podemos aumentar el batch para disminuir el tiempo del modelo.



Todos los valores elegidos en el comando utilizado de ejemplo pueden ser cambiados para encontrar la forma más eficiente de ejecución dependiendo del hardware disponible en la PC.

### 5.3 (Caso 3) Entrenamiento del Modelo YOLO con Roboflow

Roboflow ofrece integración directa con diferentes versiones de YOLO y otras arquitecturas de detección de objetos, facilitando el entrenamiento sin necesidad de configuración compleja.

La forma más sencilla de entrenar un modelo es utilizando Roboflow Train, aunque esta herramienta puede ser limitada para usuarios gratuitos. Nos da solamente 2 créditos de entrenamiento y actualmente la versión disponible es YOLOv11, aunque no es útil para nuestro caso igual la explicaremos:

- Haz click en Versions, una vez posicionados en la versión de interés, da click en Custom train.

## This version doesn't have a mode

Train an optimized, state of the art model with Roboflow or upload your model to use features like Label Assist and Model Evaluation and deploy to our auto-scaling API and edge device support.



⚡ Custom Train

- Selecciona el modelo que deseas entrenar (Roboflow 3.0 o YOLOv11)

## Roboflow Train

Automatic training on Roboflow's GPU Cluster for a [deployable](#) model within a few hours  
Available Training Credits: 2 [Upgrade Plan](#) ➔

### Select Model Architecture

#### Roboflow 3.0

- Recommended
- Fast & Accurate
- Infer with a scalable [Hosted API](#) ➔
- Deploy locally or on different devices with [Roboflow Inference](#) ➔

#### YOLOv11

- New
- Experimental
- Infer with a scalable [Hosted API](#) ➔
- Deploy locally or on different devices with [Roboflow Inference](#) ➔

Cancel

- Selecciona si buscas un modelo rápido o preciso

## Roboflow Train

Automatic training on Roboflow's GPU Cluster for a [deployable](#) model within a few hours  
Available Training Credits: 2 [Upgrade Plan](#) ➔

### Select Model Size

When training a model, there is always a trade off between inference speed and accuracy

#### Fast

- Quicker to train and infer
- Less accurate

 BASIC PLAN FEATURE

#### Accurate

- Slower to train and infer
- More accurate

Cancel

En este caso el modelo preciso es una característica de pago, solamente podremos entrenar modelos rápidos con la plataforma de Roboflow.

- Configura un “checkpoint” de entrenamiento del que se basará el modelo

### Roboflow Train

×

Automatic training on Roboflow's GPU Cluster for a [deployable](#) model within a few hours.

Available Training Credits: 2 [Upgrade Plan](#) ↗

Model Type: YOLOV11 Instance Segmentation (Fast)

#### 🚧 Train from Previous Checkpoint

Start from one of your previous training runs to speed up training and improve accuracy. This option is best if you already successfully trained a model on this project.

RECOMMENDED

#### ★ Train from Public Checkpoint

Use a pre-trained benchmark model or a starred Universe project to imbue your model with prior knowledge, reduce training time, and improve performance.

**NOTE:** Only models with the same model type as above are available as checkpoints.

Select Model

MS COCO

▼

Select Model Version

v32 - Best (Common Objects, 39.5% mAP)

▼

#### 🚧 Train from Random Initialization

Not recommended; almost always produces worse results.

Cancel

Start Training

El modelo MS COCO es un conjunto de datos a gran escala sobre detección y segmentación de objetos, lo cual es un checkpoint muy útil para cualquier entrenamiento. También podremos seleccionar modelos entrenados anteriormente.

- Haz click en “Start Training”
- Una vez finalizado, en el panel de versiones podremos ver las métricas de nuestro entrenamiento.

hojas-tbrad/1

mA  
99.1

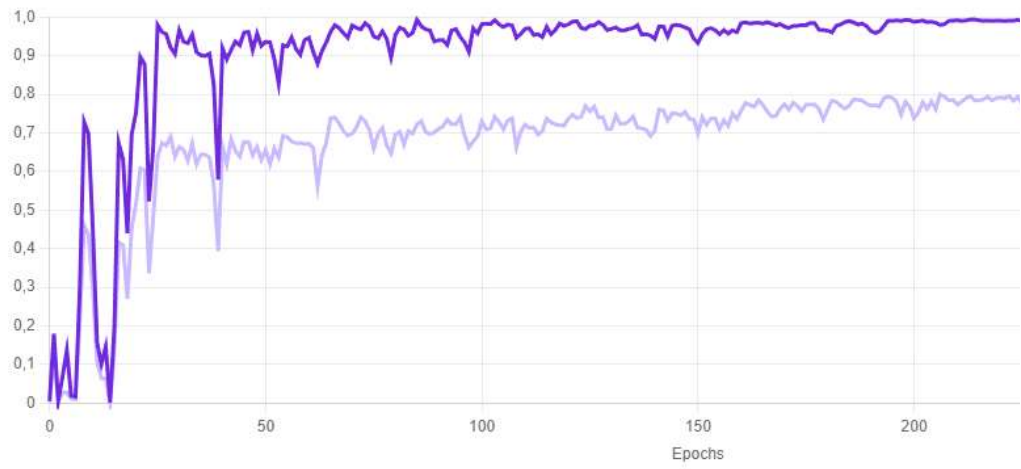
Model Type: Roboflow 3.0 Instance Segmentation (Fast)

Checkpoint: COCOOn-seg

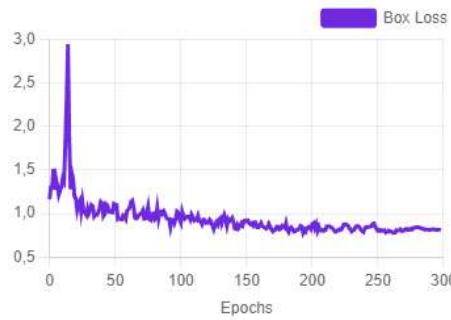
Si scrolleamos hacia abajo veremos las gráficas, como se observan en la imagen.

Training Graphs

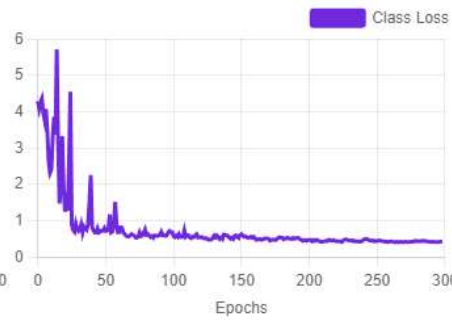
mAP



Box Loss



Class Loss



Object

