



## Guía Infraestructura - Codefest Ad Astra 2023

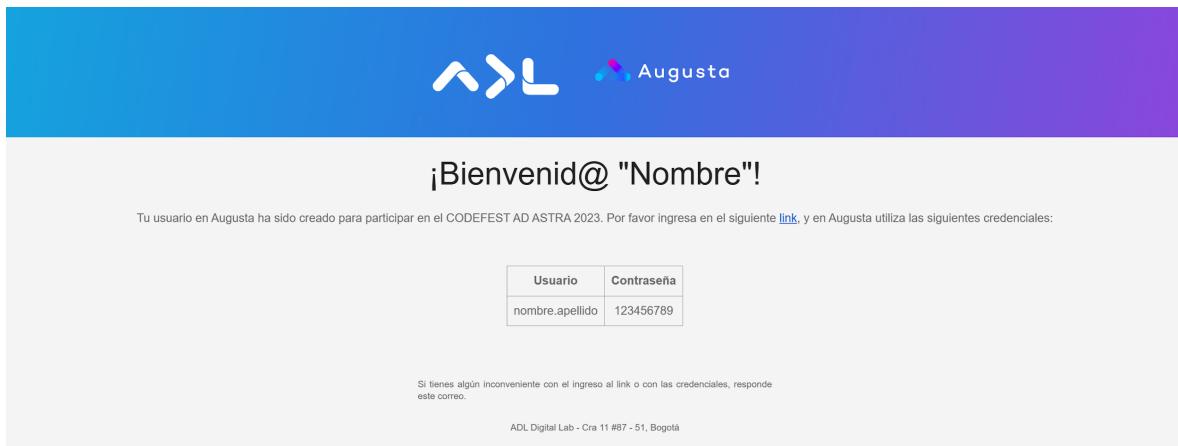
### 1. Acceso a red Unaindies

Para el evento Codefest Ad Astra 2023 contamos con la siguiente red:

**Red: FACUandes**  
**Clave: FACUandes203**

### 2. Acceso a Augusta y AWS

Se realizó el envío a todos los participantes de un correo de bienvenida en el que encontrarán el enlace de ingreso y las credenciales de acceso.



Para el acceso al entorno de desarrollo del evento, se debe ingresar a la siguiente url:  
<https://augusta-codefest.avaldigitallabs.com/>



**CODEFEST  
AD ASTRA  
2023**

Ingrese a Augusta

Augusta powered by AWS

FUERZA AÉREA COLOMBIANA

Universidad de los Andes Colombia

Software y tecnología aérea, espacial y cibernética para la protección de la amazonía

Universidad de los Andes | Departamento de Ingeniería de Sistemas y Computación | AWS | Esri | MAXAR | PERFICIENT | TELESPAZIO

Seguido de esto, hacer click en el botón *Ingreso a Augusta*, el cual va a redirigir a la página principal de AWS, solicitando las credenciales del usuario compartidas previamente en el correo de bienvenida.



#### Iniciar sesión como usuario de IAM

ID de cuenta (12 dígitos) o alias de cuenta

371243516668

Nombre de usuario:

|

Contraseña:

Recordar esta cuenta

**Iniciar sesión**

[Iniciar sesión con el email del usuario raíz](#)

[¿Olvidó la contraseña?](#)

## Amazon Lightsail

Lightsail es la manera más fácil de empezar a usar AWS

[Más información »](#)



*Codefest.adastr@fac.mil.co*



Si es la primera vez que se realiza la autenticación, el sistema automáticamente solicitará el cambio de contraseña por seguridad.

The screenshot shows the AWS IAM password change interface. At the top, there's a header with the AWS logo and a link to 'Cambio su contraseña para continuar'. Below it, the account number 'Cuenta de AWS 371243516668' is displayed. The main section is titled 'Nombre de usuario de IAM' and contains three input fields: 'Contraseña anterior' (Previous password), 'Nueva contraseña' (New password), and 'Volver a escribir la nueva contraseña' (Re-enter new password). A blue button labeled 'Confirmar cambio de contraseña' (Confirm password change) is positioned below the input fields. At the bottom of the form, there are links for 'Iniciar sesión con el email del usuario raíz' (Sign in with root user email) and a language selection dropdown set to 'Español'. A small note at the bottom states 'Términos de uso Política de privacidad © 1996-2023, Amazon Web Services, Inc. o sus empresas afiliadas.'

Luego de la autenticación y cambio de contraseña exitosa, se podrá observar la consola de aws en pantalla.

The screenshot shows the AWS Management Console homepage. The top navigation bar includes the AWS logo, a search bar, and a 'Buscar' (Search) button. Below the navigation is a banner titled 'Página de inicio de la Consola' with a 'Información' (Information) link. To the right of the banner are buttons for 'Restablecer al diseño predeterminado' (Reset to default design) and '+ Agregar widgets' (Add widgets). The main content area is divided into two sections: 'Visitados recientemente' (Recently visited) and 'Le damos la bienvenida a AWS' (Welcome to AWS). The 'Visitados recientemente' section lists services like Certificate Manager, S3, Amazon SageMaker, and Support. The 'Le damos la bienvenida a AWS' section features three cards: 'Introducción a AWS' (Introduction to AWS) with a rocket icon, 'Formación y certificación' (Training and certification) with a graduation cap icon, and '¿Cuáles son las novedades de AWS?' (What are the AWS news?) with a lightbulb icon.

### 3. Acceso al entorno de desarrollo

Los pasos para ingresar al ambiente de desarrollo son los siguientes una vez se haya realizado el inicio de sesión y se esté en la consola de Amazon Web Services:

Buscar el servicio de SageMaker



3.1 Seleccionar la región *EE.UU. Este (Ohio) us-east-2* en la consola de aws.

The screenshot shows the AWS search interface with the search term 'sage' entered. The results are categorized into Services (13), Features (27), and Resources (New). The 'Services' section is expanded, showing a list of services. The 'Amazon SageMaker' service is highlighted with a red box around its card. The card for Amazon SageMaker includes the text 'Build, Train, and Deploy Machine Learning Models'. Other visible services include Simple Queue Service (SQS), Amazon MQ, and Billing.

Category	Service	Description	
Services (13)	Amazon SageMaker	Build, Train, and Deploy Machine Learning Models	
	Simple Queue Service (SQS)	SQS Managed Message Queues	
	Amazon MQ	Managed message broker service for Apache ActiveMQ and RabbitMQ	
	Billing	Access, analyze, and control your AWS costs and usage.	
See all 13 results ▶			
Features (27)	Documentation (40,492)		
	Knowledge Articles (14)		
	Tutorials (36)		
	Events (77)		
	Marketplace (41)		
	See all 27 results ▶		
	Resources (New)	Blogs (2,791)	
		Documentation (40,492)	
		Knowledge Articles (14)	
		Tutorials (36)	
		Events (77)	
		Marketplace (41)	
		See all 13 results ▶	

3.2 Seleccionar en el menú izquierdo la opción de "Domains"



AWS Services Search [Alt+S]

IAM Amazon SageMaker Support S3

## Amazon SageMaker

Getting started Studio Studio Lab Canvas RStudio TensorBoard Domains SageMaker dashboard Images Lifecycle configurations Search ▾ JumpStart Foundation models NEW Computer vision models Natural language processing models ▾ Governance

MACHINE LEARNING

# Amazon SageMaker

Build, train, and deploy machine learning models at scale

The quickest and easiest way to get ML models from idea to production.

### How it works

What is Amazon SageMaker? Amazon SageMaker provides machine learning (ML) capabilities for data scientists to prepare, build, train, and deploy high-quality ML models efficiently.

New user onboarding guide NEW Get started with Amazon SageMaker by completing the quick start guide. Get started with SageMaker

### 3.3 Seleccionar el dominio llamado "Team-<nombre de tu equipo>"

Amazon SageMaker Dominos

Introducción Studio Laboratorio de Studio Canvas RStudio TensorBoard Domains Panel de SageMaker Imágenes Configuraciones de ciclo de vida Buscar ▾ JumpStart Foundation models NUEVO

Información

Un dominio incluye un volumen de Amazon Elastic File System (EFS) asociado, una lista de usuarios autorizados y diversas configuraciones de seguridad, aplicación, política y Amazon Virtual Private Cloud (VPC) (nube virtual privada (VPC)). Cada usuario de un dominio recibe un directorio de inicio personal y privado dentro de EFS para blocs de notas, repositorios de Git y archivos de datos.

Diagrama de la estructura de dominios

Nombre	ID	Estado	Fecha de creación	Fecha de modificación
Team-Soprote	d-mo1a5jh7kru	inService	May 24, 2023 02:04 UTC	May 24, 2023 02:09 UTC
Team-Uniandes	d-tfp5c4uezth	inService	May 24, 2023 02:04 UTC	May 24, 2023 02:11 UTC
Team-adl	d-1jyak1vnqj	inService	May 28, 2023 03:11 UTC	May 28, 2023 03:14 UTC

### 3.4 Dentro del dominio, ubicar el usuario que corresponda y hacer click en el botón "Lanzar" en la parte derecha de la pantalla y seleccionar la opción "Espacios"



Amazon SageMaker > Dominios > Dominio: Team-adl

### Team-adl

#### Detalles del dominio

Configure y administre el dominio.

Perfiles de usuarios Administración del espacio Entorno Configuración del dominio

Perfiles de usuarios Información

Un perfil de usuario representa a un único usuario dentro de un dominio, y es la principal forma de hacer referencia a un usuario a efectos de compartir, informar y otras características orientadas al usuario.

Buscar usuarios

Nombre	Fecha de modificación	Fecha de creación
participante-5	May 28, 2023 03:17 UTC	May 28, 2023 03:17 UTC
participante-4	May 28, 2023 03:16 UTC	May 28, 2023 03:16 UTC
participante-3	May 28, 2023 03:16 UTC	May 28, 2023 03:16 UTC
participante-2	May 28, 2023 03:16 UTC	May 28, 2023 03:16 UTC
participante-1	May 28, 2023 03:16 UTC	May 28, 2023 03:16 UTC

Lanzar ▲

Aplicaciones personales

- Studio
- Canvas
- TensorBoard

Colaboración

- Espacios

3.5 Se muestra un menú flotante en la pantalla para seleccionar el espacio a usar.

Se debe hacer click en el botón "Lanzar Studio"

Perfiles Información

Un perfil de usuario representa a un único usuario dentro de un dominio.

Seleccionar perfil de usuario para lanzar Studio

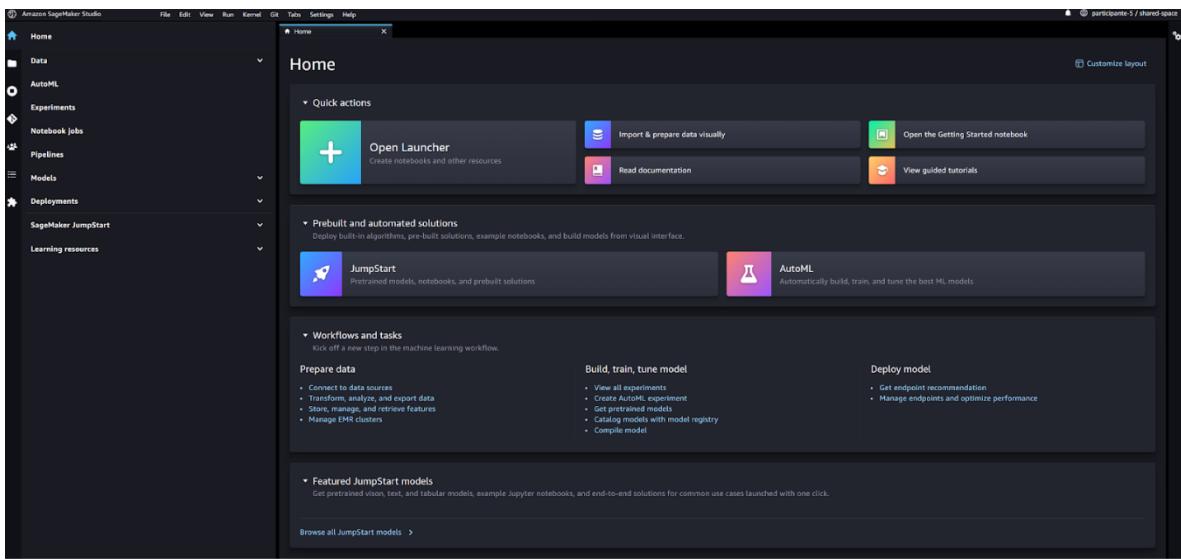
Los espacios permiten la colaboración con otros usuarios del dominio. Un espacio compartido tiene una aplicación de entorno de desarrollo integrado (IDE) común, un directorio compartido de Amazon Elastic File System (EFS) junto con el acceso concedido a todos los usuarios dentro del dominio, con lo cual es posible colaborar en tiempo real en los mismos bloques de notas y otros recursos.

Buscar recursos

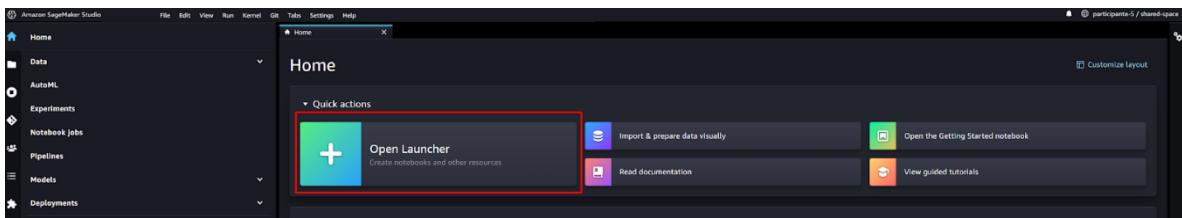
Nombre	Rol de ejecución
shared-space	arn:aws:iam::371243516668:role/Sagemaker-codefest-execution-user-role

Lanzar Studio

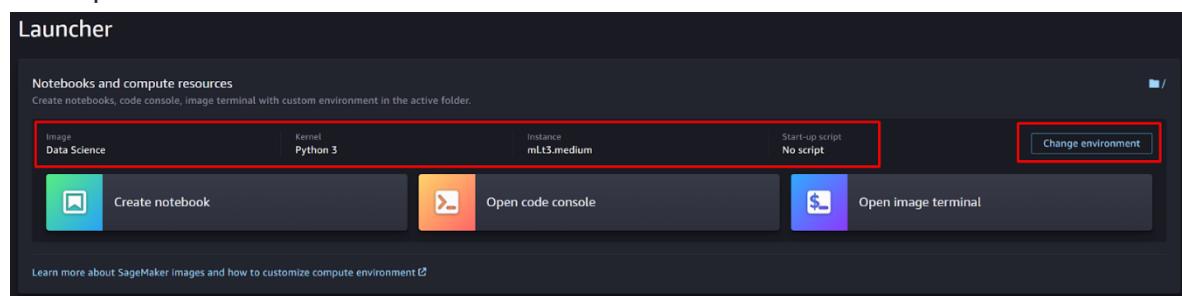
3.6 El ambiente se demora unos minutos en iniciar la primera vez. Una vez cargado, se despliega Amazon SageMaker Studio, el cual es un ambiente de JupyterLab.



3.7 Para iniciar por primera vez algún notebook y disponibilizar el cómputo que lo va a soportar, se debe hacer click en la opción de "Open Launcher"



3.8 Hacer click en el botón de "Change environment" y configurarlo según lo que se requiera.



3.9 En el menú flotante que se muestra en la pantalla están los siguientes campos:

Image: imagen del contenedor que se va a utilizar para el ambiente.

Documentación de las imágenes disponibles en sagemaker:



<https://docs.aws.amazon.com/sagemaker/latest/dg/notebooks-available-images.html>

Tips: Algunas imágenes tienen versiones de python más avanzadas o cuentan con una base para trabajar con GPU

Kernel: ambientes de python que se tengan disponibles (seleccionar el que está por defecto)

Instance type: tipo de instancia a utilizar para el cómputo.

Para el reto se tienen disponibles las siguientes máquinas:

Tipo: ml.g4dn.2xlarge

vCPU: 8

RAM: 32

GPU: 1 tipo T4

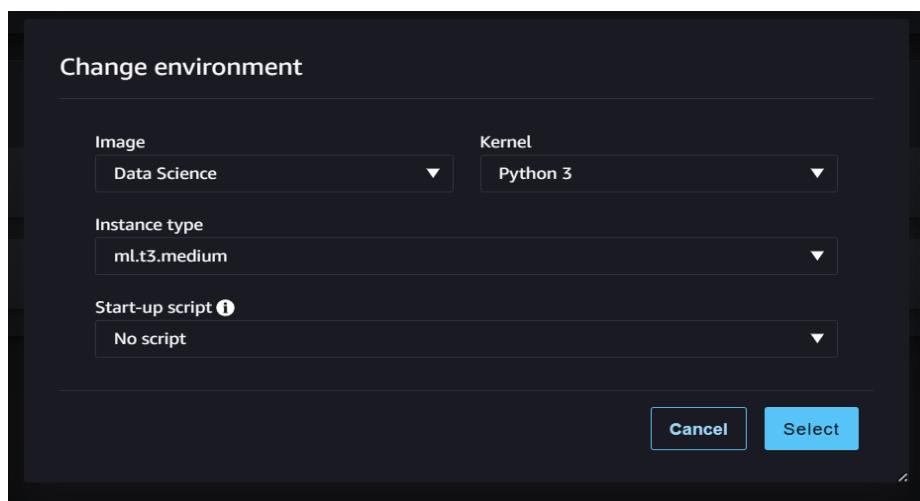
Tipo: ml.m5d.4xlarge

vCPU: 16

RAM: 64

GPU: N/A

**Start-up script:** N/A



3.10 Una vez seleccionados los valores que se desean, se observa el menú con la nueva configuración. Hacer click en "Create notebook" para desplegar el servidor y el notebook asociado.



Launcher

Notebooks and compute resources  
Create notebooks, code console, image terminal with custom environment in the active folder.

Images Data Science 3.0 Kernel Python 3 Test image ml.m5.large Start up script No script Change environment

Create notebook Open code console Open image terminal

Learn more about SageMaker images and how to customize compute environment ↗

Utilities and files System terminal Text file Markdown file Python file Notebook jobs Contextual help

3.11 La creación del servidor tarda pocos minutos la primera vez, luego se pueden añadir más notebooks de forma inmediata si es usada la misma configuración.

3.12 ¡En este momento el notebook está listo para ser utilizado!

Se recomienda acceder a la siguiente documentación para conocer y profundizar en el entorno de desarrollo de SageMaker:

<https://docs.aws.amazon.com/sagemaker/latest/dg/studio-ui.html>



## 4. Funciones o comandos útiles

### 4.1 Repositorios de datos para el evento:

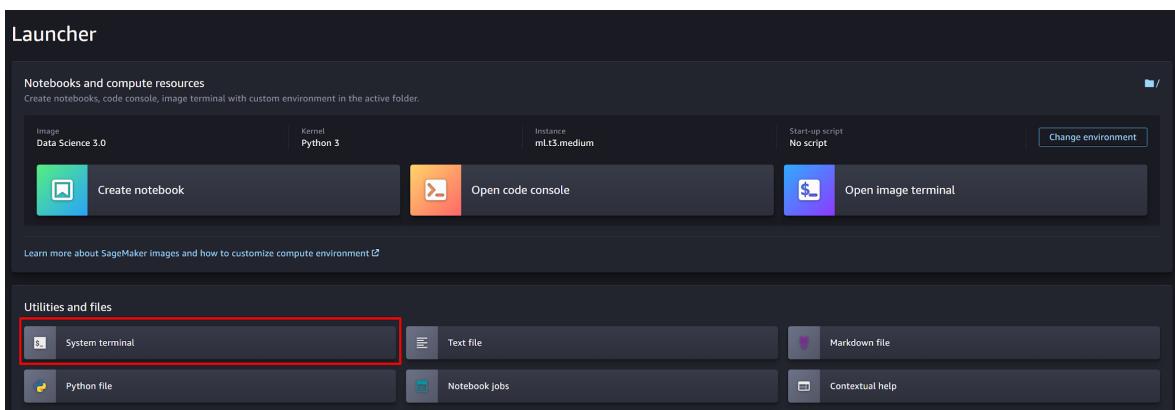
El repositorio oficial para la información que dispone el codefest es un bucket de S3 llamado *codefest-data*. Para el acceso a este bucket, se cuenta con el permiso de lectura, proporcionado únicamente con el ingreso desde la herramienta de SageMaker.

Dentro del bucket actualmente se encuentran las siguientes rutas con archivos de pruebas: *Imágenes, Videos, Textos, Scripts*

Adicionalmente a este bucket, cada equipo cuenta con uno propio para poder guardar los archivos que requieran. El nombre del bucket tiene la siguiente estructura: *codefest-team-<nombre del equipo>*. Por ejemplo, si el equipo se llama “Los halcones”, el nombre del bucket es *codefest-team-los-halcones* (todo en minúscula y reemplazando posibles espacios por guiones medios)

### 4.2 Ejecutar una terminal en el ambiente de SageMaker:

Para realizar lo anterior, hacer click en la opción “System terminal”, que se encuentra en la opción de Launcher (ver la sección 2.7).





Seguidamente, se carga la terminal en una pestaña nueva en la cual podremos ejecutar comandos a nivel de sistema operativo, como se observa en la siguiente imagen:

```
!!!!!! Welcome to SageMaker Studio System Terminal !!!!!!!

Below are some useful tips:

* Activate studio conda environment using "conda activate studio" to install extensions, list extension etc. For more information, see https://docs.aws.amazon.com/sagemaker/latest/dg/studio-jl.html#studio-jl-install
* Post JupyterServer extension installation, if needed, restart just the server(not app) using "restart-jupyter-server"

sagemaker-user@studio: ~
```

4.3 Si no se ha podido encontrar el bucket del equipo, se puede ejecutar el comando **aws s3 ls** en la terminal del ambiente y obtener el listado de todos los buckets disponibles, como podemos observar en la siguiente imagen:

```
!!!!!! Welcome to SageMaker Studio System Terminal !!!!!!!

Below are some useful tips:

* Activate studio conda environment using "conda activate studio" to install extensions, list exte
* Post JupyterServer extension installation, if needed, restart just the server(not app) using "re

sagemaker-user@studio$ aws s3 ls
2023-05-24 02:13:12 codefest-data
2023-05-29 23:45:28 codefest-team-adl
2023-05-27 05:46:28 codefest-team-aero-code
2023-05-27 05:46:31 codefest-team-amazonia-guard-tech
2023-05-27 05:46:31 codefest-team-amazontech
2023-05-27 05:46:31 codefest-team-ara
2023-05-27 05:46:28 codefest-team-astroamazonicos
2023-05-27 05:46:33 codefest-team-byte-busters
2023-05-27 05:46:33 codefest-team-c3d
2023-05-27 05:46:28 codefest-team-co2scientes
2023-05-27 05:46:31 codefest-team-codigo-aberto-del-amazonas
2023-05-27 05:46:33 codefest-team-condor
2023-05-27 05:46:28 codefest-team-debugging-dynamics
2023-05-27 05:46:31 codefest-team-deep-mind-mavericks
2023-05-27 05:46:28 codefest-team-eagleandes
2023-05-27 05:46:28 codefest-team-eco-coders
2023-05-27 05:46:28 codefest-team-ecovision
2023-05-27 05:46:31 codefest-team-gamma-ai
2023-05-27 05:46:33 codefest-team-geo-army
```

4.4 Para cargar o descargar archivos entre los buckets y el ambiente local, utilizando la terminal mencionada anteriormente, se pueden implementar los siguientes comandos:

**Cargar archivo específico:**

```
aws s3 cp archivo_local s3://codefest-team-<nombre del equipo>/
```

**Sincronizar (cargar) carpeta:**

```
aws s3 sync carpeta_local/ s3://codefest-team-<nombre del equipo>/
```

**Descargar archivo específico:**

```
aws s3 cp s3://codefest-team-<nombre del equipo>/archivo_bucket archivo_local
```

**Sincronizar (descargar) carpeta:**

```
aws s3 sync s3://codefest-team-<nombre del equipo>/ carpeta_local/
```

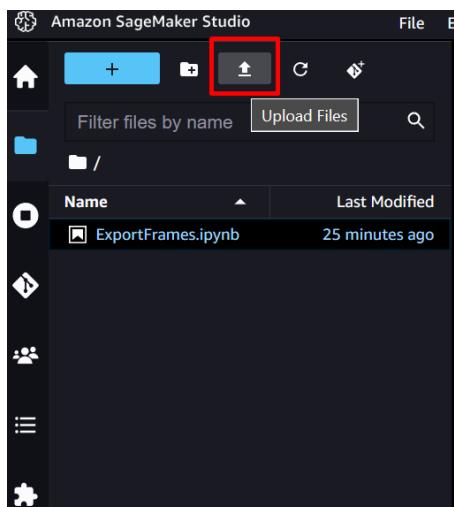
Para mayor información de los comandos de *cp* y *sync*, se puede ejecutar la ayuda de aws en la terminal de sagemaker del paso anterior, a través de los siguientes comandos:

```
aws s3 cp help  
aws s3 sync help
```

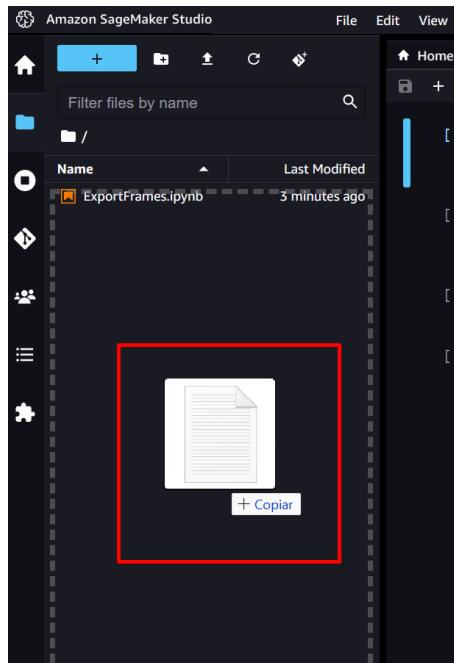
4.5 Para cargar y descargar archivos directamente desde el entorno de SageMaker, podemos realizar lo siguiente:

**Para carga de archivos:**

Hacer click en la opción “Upload files” del entorno de SageMaker y seleccionar el archivo que queremos cargar.

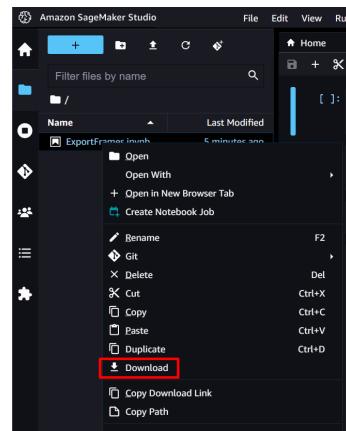


Alternativamente, se encuentra disponible la carga de archivos arrastrando directamente al espacio de objetos del entorno.



### Para descarga de archivos:

Para descargar el archivo (que ya se encuentra en el ambiente de Sagemaker) se debe hacer click derecho en dicho archivo y escoger la opción *Download*:





4.6 Para cargar un archivo hacia el bucket de S3, a través del lenguaje de programación Python, podemos utilizar el siguiente código:

```
import boto3
s3 = boto3.client('s3')
s3.upload_file('/ruta_archivo/local.ext', 'bucket', 'nombre_archivo_destino.ext')
```

4.7 Para descargar un archivo desde el bucket de S3, a través del lenguaje de programación Python, podemos utilizar el siguiente código:

```
import boto3
s3 = boto3.client('s3')
s3.download_file('bucket', 'ruta_archivo_bucket', 'nombre_local.ext')
```

## 5. Retos Videos y Texto

### 4.1. Consideraciones generales

#### 4.1.1 Instalación de paquetes python y linux.

Cuando necesites instalar librerías de python o paquetes de linux como por ejemplo libgl1 necesaria para usar opencv, debes anteponer el simbolo ! en tu celda de trabajo:

```
!apt-get update --yes && apt-get install libgl1 --yes
!pip3 install boto3 opencv-python
```

Nota: Recuerda que tu ambiente de trabajo para procesamiento de imágenes tiene una versión de python 8 y pytorch como framework de trabajo para NLP y computer visión.

#### 4.1.2 Verificación de GPU

Si tu modelo es demasiado grande, tienes disponible un entorno con GPU. Podrás verificar que esté activa de la siguiente manera:

```
import torch
torch.cuda.is_available()
```

True = Disponible



False = No disponible

Una vez confirmada su disponibilidad debes setearla, de la siguiente manera, usando torch:

```
device: str = "cuda" if torch.cuda.is_available() else "cpu"
```

```
import torch.nn as nn
import torch.nn.functional as F

class Net(nn.Module):

    def __init__(self, device : str =device):

        super(Net, self).__init__()
        # 1 input image channel, 6 output channels, 5x5 square convolution
        # kernel
        self.conv1 = nn.Conv2d(1, 6, 5)
        self.conv2 = nn.Conv2d(6, 16, 5)
        # an affine operation: y = Wx + b
        self.fc1 = nn.Linear(16 * 5 * 5, 120) # 5*5 from image dimension
        self.fc2 = nn.Linear(120, 84)
        self.fc3 = nn.Linear(84, 10)
```

## 4.2. Análisis de imágenes

### 4.2.1. Procesamiento de datos

#### 4.2.1.1 Generación de imágenes a partir de un video:

Puedes tratar el video como una serie de frames, los cuales se tomarán según sea tu criterio (x por segundo). Te sugerimos utilizar las siguientes herramientas opencv-python o ffmpeg de linux. Este último puede ser útil ya que es levemente más rápido en la conversión.

```
!pip3 install opencv-python
!apt-get update --yes && apt-get install ffmpeg --yes
```



Si lo requieres puedes aplicar filtros sobre las imágenes. Hay librerías como PIL las cuales poseen filtros que podrían ser de tu interés.

#### 4.2.1.2 Generación de anclas:

La detección de objetos mediante redes neuronales de aprendizaje profundo puede proporcionar un medio rápido y preciso para predecir la ubicación y el tamaño de un objeto en una imagen. Idealmente, la red devuelve objetos válidos de manera oportuna, independientemente de la escala de los objetos. El uso de cuadros de anclaje mejora la velocidad y la eficiencia de la parte de detección de un marco de red neuronal de aprendizaje profundo.

¿Qué es una caja de ancla?

Los cuadros de anclaje son un conjunto de cuadros delimitadores predefinidos de una cierta altura y anchura. Estos cuadros se definen para capturar la escala y la relación de aspecto de las clases de objetos específicos que desea detectar y, por lo general, se eligen según los tamaños de los objetos en sus conjuntos de datos de entrenamiento. Durante la detección, los cuadros de anclaje predefinidos se distribuyen en mosaico a lo largo de la imagen. La red predice la probabilidad y otros atributos, como el fondo, la intersección sobre la unión (IoU) y las compensaciones para cada cuadro de anclaje en mosaico. Las predicciones se utilizan para refinar cada cuadro de anclaje individual. Puede definir varios cuadros de anclaje, cada uno para un tamaño de objeto diferente. Los cuadros de anclaje son conjecturas de cuadro de límite inicial fijo.

La red no predice directamente los cuadros delimitadores, sino que predice las probabilidades y los refinamientos que corresponden a los cuadros ancla en mosaico. La red devuelve un conjunto único de predicciones para cada cuadro de anclaje definido. El mapa de características final representa las detecciones de objetos para cada clase. El uso de cuadros de anclaje permite que una red detecte múltiples objetos, objetos de diferentes escalas y objetos superpuestos.<sup>1</sup>

Una de las técnicas usadas para calcular anclas es a través de centroides k-means y un algoritmo genético.

---

<sup>1</sup> <https://www.mathworks.com/help/vision/ug/anchor-boxes-for-object-detection.html>



Esta es una pequeña función que te puede ayudar si deseas generar anclas, ten en cuenta que estas son usadas en modelos como yolo, RetinaNet y Faster R-CNN.

```
import random

import torch
import numpy as np
from scipy.cluster.vq import kmeans
import warnings


def generate_anchors(dataset, n=9, img_size=416, thr=4.0, gen=1000, verbose=True):
    """ Creates kmeans-evolved anchors from training dataset

    Arguments:
        dataset: Diccionario que contiene, arreglo con el tamaño de todas las imágenes de entrenamiento, lista con las etiquetas y recuadros por imagen.
        n: numero de anclas que quieres generar
        img_size: tamaño de la recuadro maximo usado para entrenar
        thr: anchor-label wh. Ancho del radio para calcular el kmeans. Default=4.0
        gen: generaciones para evolucionar anclas usando algoritmo genético
        verbose: imprimir los resultados

    Return:
        k: Arreglo con las anclas calculadas. El número varia dependiendo de su elección
    """
    print("generando anclas")
    thr = 1 / thr

    def metric(k, wh): # compute metrics
        r = wh[:, None] / k[None]
        x = torch.min(r, 1 / r).min(2)[0] # ratio metric
        return x, x.max(1)[0] # x, best_x

    def anchor_fitness(k): # mutation fitness
        _, best = metric(torch.tensor(k, dtype=torch.float32), wh)
        return (best * (best > thr).float()).mean() # fitness

    def print_results(k, verbose=True):
        k = k[np.argsort(k.prod(1))] # sort small to large
        if verbose:
            x, best = metric(k, wh0)
            bpr, aat = (best > thr).float().mean(), (x > thr).float().mean() * n # best
possible recall, anch > thr
```



```
s = f'thr={thr:.2f}: {bpr:.4f} best possible recall, {aat:.2f} anchors past thr\n' \
      f'n={n}, img_size=img_size, metric_all={x.mean():.3f}/{best.mean():.3f}-mean/best, ' \
          f'past_thr={x[x > thr].mean():.3f}-mean: '
    print(s)
  return k

# Get label wh
shapes = img_size * dataset["shapes"] / np.array(dataset["shapes"]).max(1,
keepdims=True)

wh0 = np.concatenate([l[:, 3:5] * s for s, l in zip(shapes, dataset["labels"])]) # wh

# Filter
i = (wh0 < 3.0).any(1).sum()
if i and verbose:
    print(f'WARNING: Extremely small objects found. {i} of {len(wh0)} labels are < 3
pixels in size.')
wh = wh0[(wh0 >= 2.0).any(1)] # filter > 2 pixels
# wh = wh * (np.random.rand(wh.shape[0], 1) * 0.9 + 0.1) # multiply by random scale 0-1

# Kmeans calculation
s = wh.std(0) # sigmas for whitening
k, dist = kmeans(wh / s, n, iter=30) # points, mean distance
assert len(k) == n, f'ERROR: scipy.cluster.vq.kmeans requested {n} points but returned
only {len(k)}'
k *= s
wh = torch.tensor(wh, dtype=torch.float32) # filtered
wh0 = torch.tensor(wh0, dtype=torch.float32) # unfiltered
k = print_results(k, verbose=False)

# Evolve
npr = np.random
f, sh, mp, s = anchor_fitness(k), k.shape, 0.9, 0.1 # fitness, generations, mutation
prob, sigma
if verbose:
    print("Generating anchor boxes for training images...")
for _ in range(gen):
    v = np.ones(sh)
    while (v == 1).all(): # mutate until a change occurs (prevent duplicates)
        v = ((npr.random(sh) < mp) * random.random() * npr.randn(*sh) * s + 1).clip(0.3,
3.0)
    kg = (k.copy() * v).clip(min=2.0)
    fg = anchor_fitness(kg)
```



```
    if fg > f:
        f, k = fg, kg.copy()
    return k
```

#### 4.2.1.2 Preparación de imágenes para la red

Las siguientes funciones te pueden ser útiles para ajustar el tamaño de la imagen a tamaño de la red y almacenarla en un tensor.

```
def prepare_image(
    image : np.ndarray,
    inp_dim : Tuple[int, int]) -> torch.Tensor:
    """
    Preparar la imagen para que coincida con la red
    """
    print("1. PREPARANDO LA IMAGEN")
    print("1.1 Ajustar la imagen con el tamaño de la red")
    img = letterbox_image(image, inp_dim)
    print("1.2 se hace un transpose del arreglo")
    img = img[:, :, ::-1].transpose((2, 0, 1)).copy()
    print("1.3 se asigna un arreglo a un sensor")
    img = torch.from_numpy(img).float().div(255.0).unsqueeze(0)
    return img

def letterbox_image(
    image : np.ndarray,
    inp_dim : Tuple[int, int]) -> np.ndarray:
    """
    Cambia el tamaño de las imágenes a la dimensión esperada por la red.
    """
    print("1.1 Proceso que cambia el tamaño (dimensión de la imagen por la esperada por la red.)")
    img_w, img_h = image.shape[1], image.shape[0] # original image dimension
    net_w, net_h = inp_dim # the dimension expected by the network.

    scale_factor = min(net_w/img_w, net_h/img_h)
    new_w = int(round(img_w * scale_factor))
    new_h = int(round(img_h * scale_factor))

    resized_image = cv.resize(image, (new_w, new_h), interpolation=cv.INTER_CUBIC)
```



```
canvas = np.full((net_w, net_h, 3), 128)
    canvas[(net_h - new_h)//2 : (net_h - new_h)//2 + new_h, (net_w - new_w)//2 : (net_w - new_w)//2 + new_w, :] = resized_image
return canvas
```

#### 4.2.1.4 Lectura de las etiquetas y asignación a un tensor

Si tienes en un archivo plano tus etiquetas y los boxes, la siguiente función te podrá guiar para leerlos y asignarlos a un tensor. La función unifica si por cada imagen se generó un archivo plano.

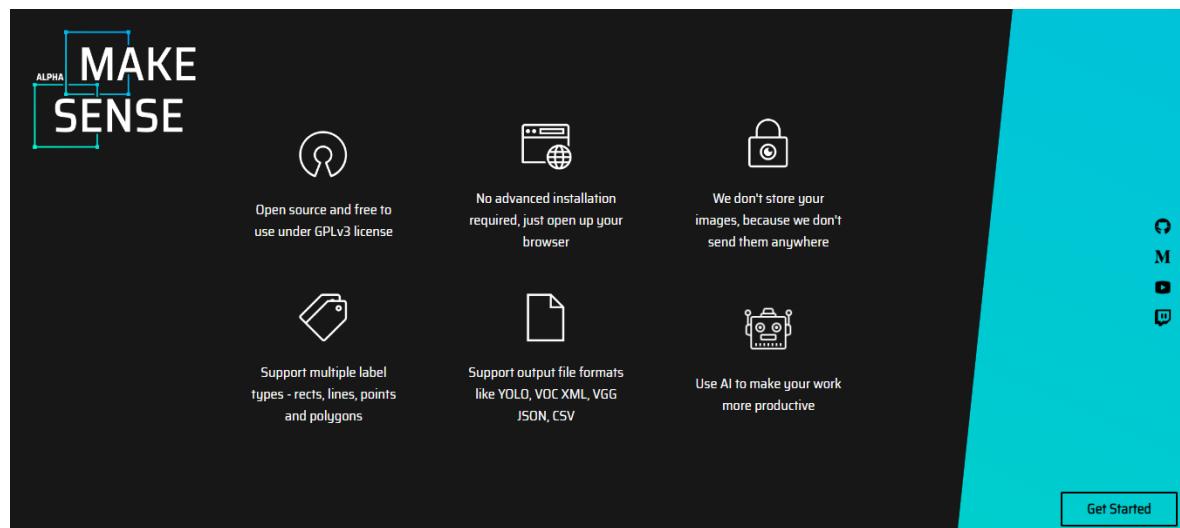
```
def load_annotations(self):
    annotations = []
    annotation_files = os.listdir(self.annotation_dir)
    for annotation_file in annotation_files:
        image_id = os.path.splitext(annotation_file)[0]
        image_path = self.data_path + image_id + '.jpg'
        try:
            with open(os.path.join(self.annotation_dir, annotation_file), 'r') as file:
                lines = file.readlines()
            boxes = []
            labels = []
            for line in lines:
                data = line.strip().split(' ')
                label,x_min, y_min, x_max, y_max = map(float, data)
                boxes.append([x_min, y_min, x_max, y_max])
                labels.append(int(label))
            annotations.append({
                'image_path': image_path,
                'boxes': boxes,
                'labels': labels
            })
        except:
            pass
    return annotations
```

#### 4.2.2. Marcación de objetos



Debes definir la forma de modelar. Si deseas hacer detección de objetos te proponemos para taggear, la siguiente página:

<https://www.makesense.ai/>



En ella encontrarás el botón “Get Started” y a continuación se te presentará una pa

Nota: si vas a utilizar modelos pre entrenados recuerda que deben ser compatibles con torch (extensión .pt, pth, etc)

#### 4.2.3. Sugerencias en la construcción de modelos.

- Puedes guiarte en técnicas como yolo, tini-yolo para la detección de objetos.
- En el diseño de tu red al correr las épocas te recomendamos ir guardando los modelos resultantes por cada época. Así no sea el modelo definitivo esto te servirá como insumo para construir el módulo de predicción.
- Ten en cuenta que la selección de las épocas y su tasa de aprendizaje influirá en la velocidad de entrenamiento.
- Puedes utilizar un modelo preentrenado, pero ten en cuenta que hay etiquetas específicas que solo lograrás encontrar entrenando un modelo.
- En el caso de los vehículos puedes unificar conceptos parecidos, por ejemplo maquinaria amarilla con dragas. Esto te ayudará a mejorar el rendimiento si se tienen muy pocos objetos de entrenamiento (etiquetas).



### 4.3. Análisis de texto

El Named-entity recognition (NER) es una subtarea de extracción de información que busca localizar y clasificar entidades nombradas mencionadas en textos no estructurados dentro de categorías predefinidas como: nombres de personas, organizaciones, ubicaciones, códigos médicos, expresiones de tiempo, cantidades, valores monetarios, porcentajes, etc.

#### 4.3.1. Procesamiento de datos

Para facilitar el uso de las librerías y extensiones para el análisis de texto y desarrollo del reto se sugiere seleccionar el siguiente ambiente:

Set up notebook environment

Set up environment for "Prueba\_NER.ipynb".

Image	Kernel
PyTorch 1.10 Python 3.8 CPU Opti...	Python 3
Instance type	
ml.t3.medium	
Start-up script ⓘ	
No script	

Y hacer la instalación de la librería *transformers*:

```
!pip install transformers
Collecting transformers
  Using cached transformers-4.29.2-py3-none-any.whl (7.1 MB)
```

#### 4.3.2. Marcación de texto

*Codefest.adastr@fac.mil.co*



#### 4.3.2.1 Usando NER Text Annotator

Es posible usar una herramienta de libre acceso como *NER Text Annotator*. Enlace [aqui](#).

The screenshot shows the NER Text Annotator software interface. On the left, there are configuration options: 'Text Separator' set to 'New Line', 'Annotation Precision' set to 'Word Level', and 'Tagging Progress (5/22)'. On the right, the main window displays a text document with several entities annotated with colored boxes and tags: 'GOB' (red), 'LOC' (blue), 'PER' (green), and 'OTHER' (orange). The text reads: 'preocupante incremento de los cultivos de coca OTHER en el amazonas LOC de acuerdo al reporte correspondiente a estudios del año 2016 presentado por la oficina de naciones unidas contra la droga y el delito GOB unodc en el cual el balance del sistema de monitoreo de cultivos ilícitos OTHER simci'.

En esta página podrán cargar el archivo de texto (en formato .txt) preferiblemente sobre el conjunto de textos ya depurados, la salida será un archivo .json que podrán cargar dentro del espacio de trabajo.

#### 4.3.3 Sugerencias para realizar el modelo de reconocimiento de entidades (NER)

- Es buena idea iniciar con un modelo open source pre-entrenado para esta finalidad. A diferencia de las tareas relacionadas con imágenes, los modelos entrenados con diferentes idiomas afectan considerablemente los resultados esperados de un idioma a otro. Verifica que el modelo haya sido pre-entrenado en el idioma español o sea multilenguaje.
- Evalúe el desempeño del modelo con una muestra de textos
- Si consigue los resultados esperados, puede iniciar un ajuste fino del modelo con marcaciones de textos propios del reto o algún dataset público. Con esto, posiblemente puede: 1) obtener aumento de la cantidad de entidades (por ejemplo FECHA, CANTIDAD, etc), 2) ser más específico en algunas entidades (ejemplo: LOC-DEPARTAMENTO, LOC-PAIS, etc) o 3) mejorar los resultados.
- También es posible que opte por un entrenamiento completo, en este caso es importante considerar la cantidad de texto disponible marcado. Algunas preguntas a tener en cuenta son: ¿Es suficiente el texto suministrado para el reto? ¿Es necesario conseguir más información pública?, si este texto no está marcado, ¿Cuánto tiempo tomaría marcarlo?
- Es posible que considere utilizar un gran modelo de lenguaje open-source de uso general, si es así, posiblemente tendrías que suministrar algunos ejemplos (few shot



learning), evalúa el desempeño de esta opción y asegúrese de no compartir información sensible.

#### 4.3.4 Sugerencias para realizar el modelo de clasificación de textos en categorías:

- Para realizar modelos de clasificación con texto primero se debe tener en cuenta el preprocesamiento y limpieza de los datos, en ese sentido es recomendable revisar si los textos contienen símbolos, tildes, espacios y estandarizar las mayúsculas.
- Hay muchas palabras que no aportan ningún tipo de valor a la hora de clasificar un texto y por el contrario si añaden complejidad, es recomendable remover las stopwords con librerías que ya se han construido para este fin.
- Continuando con la reducción de complejidad, es recomendable lematizar las palabras. Un ejemplo de esto es mesa y mesas, son diferentes palabras pero con el mismo lema mesa.
- Se recomienda hacer algún tipo de visualización de los datos ya que esto da una perspectiva un poco más amplia de los corpus que se tienen.
- La estructura que se requiere para construir la mayoría de los modelos de clasificación siguen el formato de bolsa de palabras o “Bag Of Words” en el cual cada documento representa una fila del data frame y cada palabra representa una columna, esto se puede lograr a través de tokenizadores y contadores.
- Al igual que en la mayoría de los modelos de clasificación se recomienda guardar una muestra de entrenamiento y otra de validación para evitar problemas de overfitting, adicionalmente se recomienda utilizar algún tipo de muestreo estratificado cuando la proporción de las categorías es desbalanceada.
- En términos de modelos de clasificación o multi clasificación no hay ningún tipo de restricción y se recomienda hacer un tuneo de parámetros para obtener las mejores métricas posibles.

### 5. Restricciones y limitaciones

- No se permite usar apis externas en la entrega definitiva.
- La solución solo es aceptada por los jurados si el desarrollo final se entrega a través de la plataforma Augusta.

### 6. Recomendaciones

- Investigue cuáles soluciones de modelos pre-entrenados open source existen en internet.



- Pruebe con una imagen o parte del texto si estas soluciones resuelven el problema propuesto en el codefest.
- Si es así, revise si la solución requiere fine tuning (entrenar parte del modelo) o está lista para la predicción.
- Tenga en cuenta que es posible que encuentre una arquitectura o algoritmo que resuelva un problema similar pero requiere un entrenamiento completo.
- Procure estimar la cantidad de horas/cómputo en las máquinas disponibles con una muestra (algunas imágenes o parte del texto), luego calcule cuánto tiempo le tomará la inferencia o entrenamiento completo. Esto le dará una idea de si la capacidad de cómputo disponible es suficiente.
- Recuerde que algunos modelos tienen diferentes tamaños de parámetros, entonces es posible que tenga que sacrificar tamaño por tiempo de cómputo y aún así obtener un resultado similar.