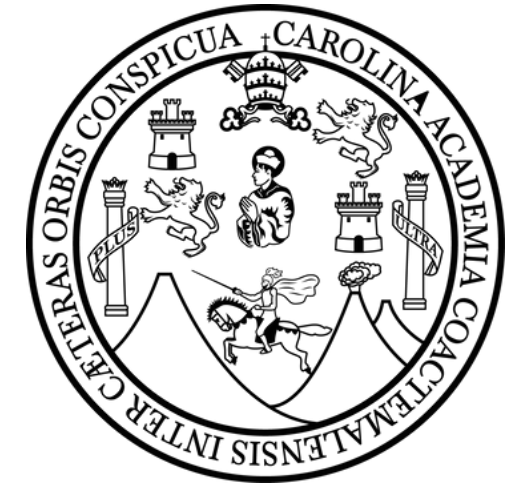


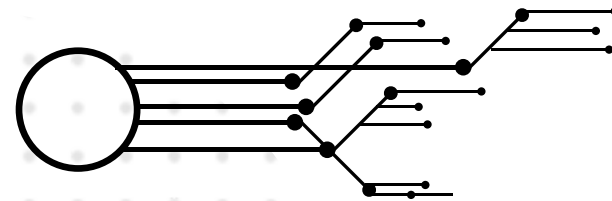


IA EN ASTROFÍSICA



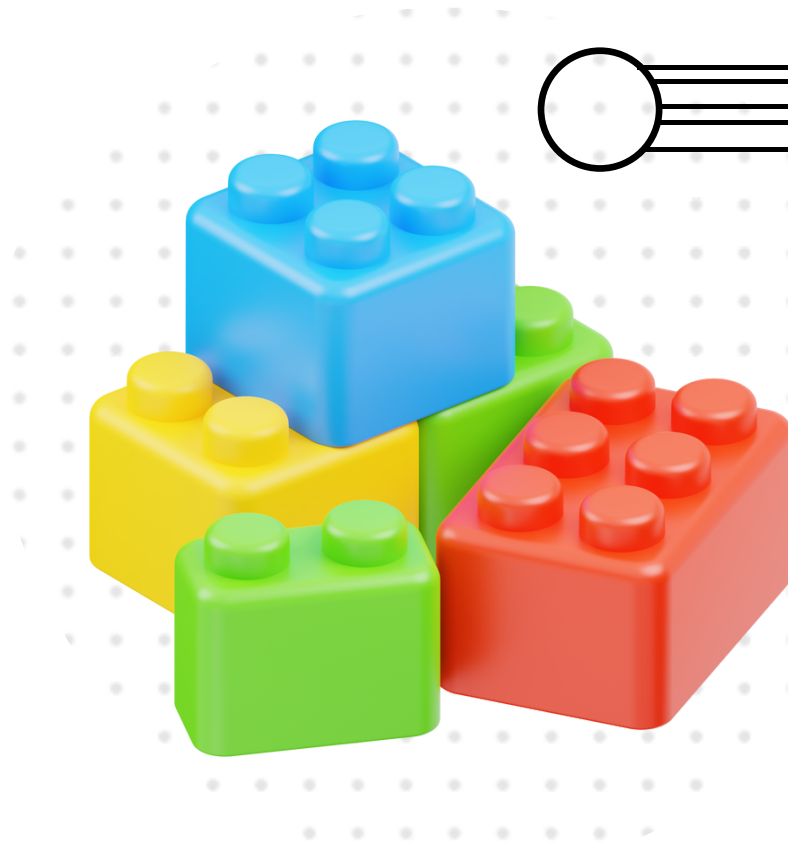
TAACO 2024

Clase 3:



Convoluciones y redes adversariales

Primeras aplicaciones a la astrofísica



Pedro Elías Mirón Enríquez (pmiron@astro.unam.mx)

20 de noviembre de 2024, USAC, Ciudad de Guatemala

CONVOLUCIÓN

La convolución es una operación matemática que calcula el área de superposición entre una función fija y otra que se invierte y traslada, actuando como un promedio móvil generalizado.

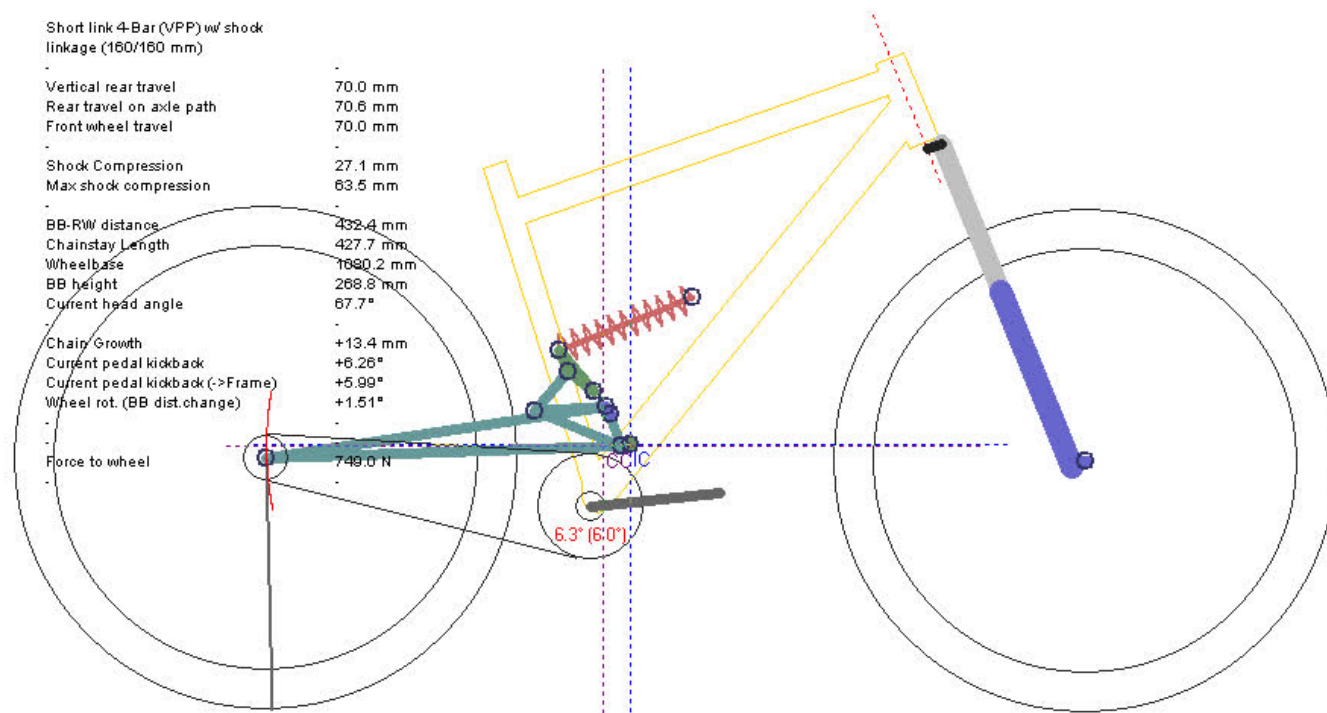
$$(f * g)(t) = \int_{-\infty}^{\infty} f(\tau)g(t - \tau)d\tau$$



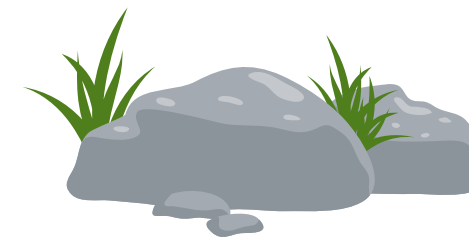
Gato Huh, [giftenor](#)

CONVOLUCIÓN: ANALOGÍA

- La función que describe a la suspensión
- La función que describe el camino lleno de rocas
- ¿Qué es la convolución?



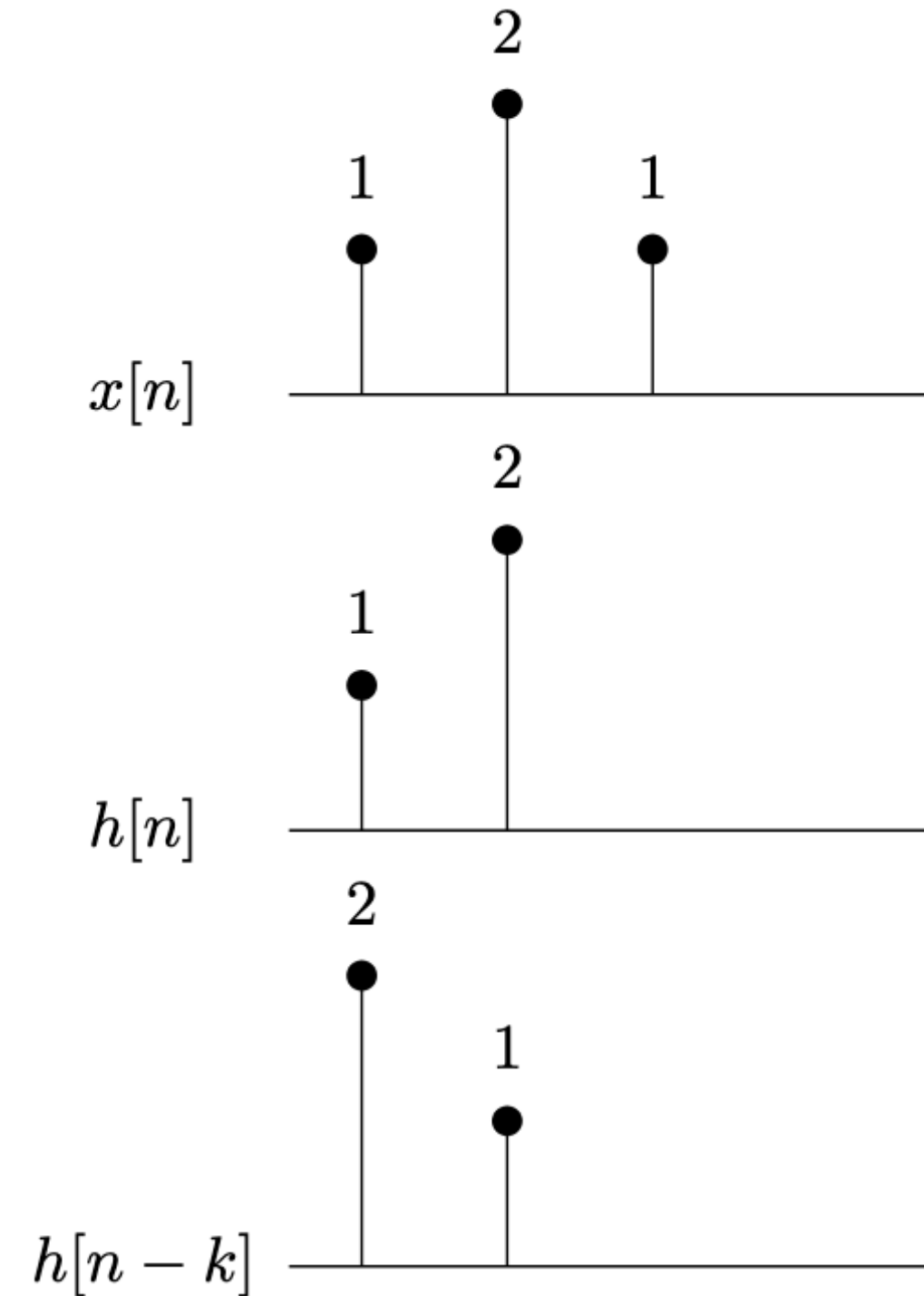
Fuente: chazdog, modelando su idea para modificar su bicicleta en www.pinkbike.com



CONVOLUCIÓN: 1D

$$x[n] = \{1, 2, 1\} \quad h[n] = \{1, 2\}$$

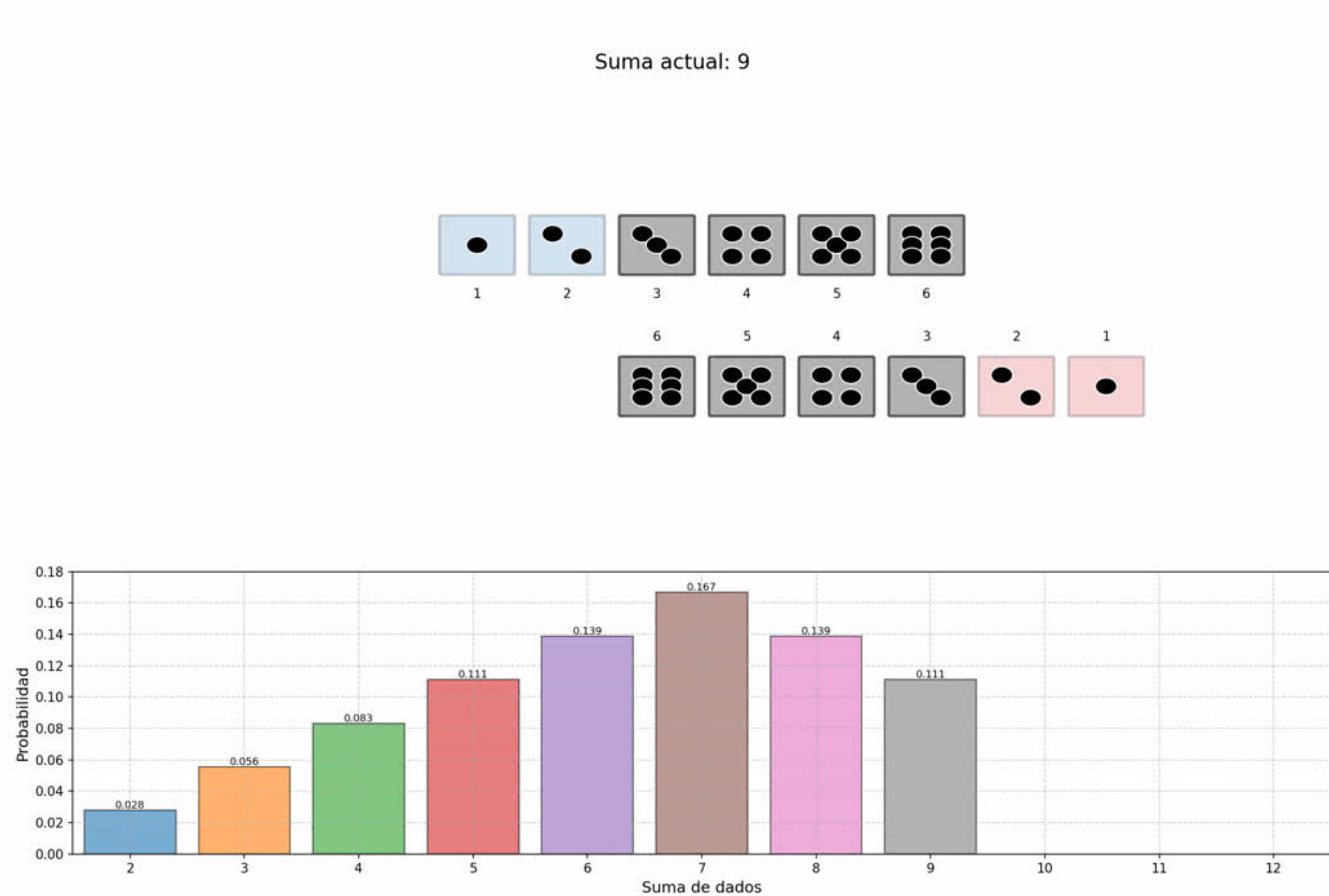
$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$



CONVOLUCIÓN Y PROBABILIDAD

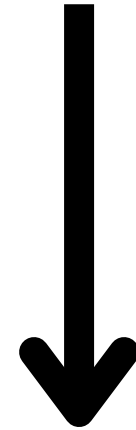
Supongamos que tenemos dos eventos independientes, como tirar dos dados, **¿Cómo calculamos la probabilidad de las sumas de estos dados?**

CONVOLUCIÓN Y PROBABILIDAD



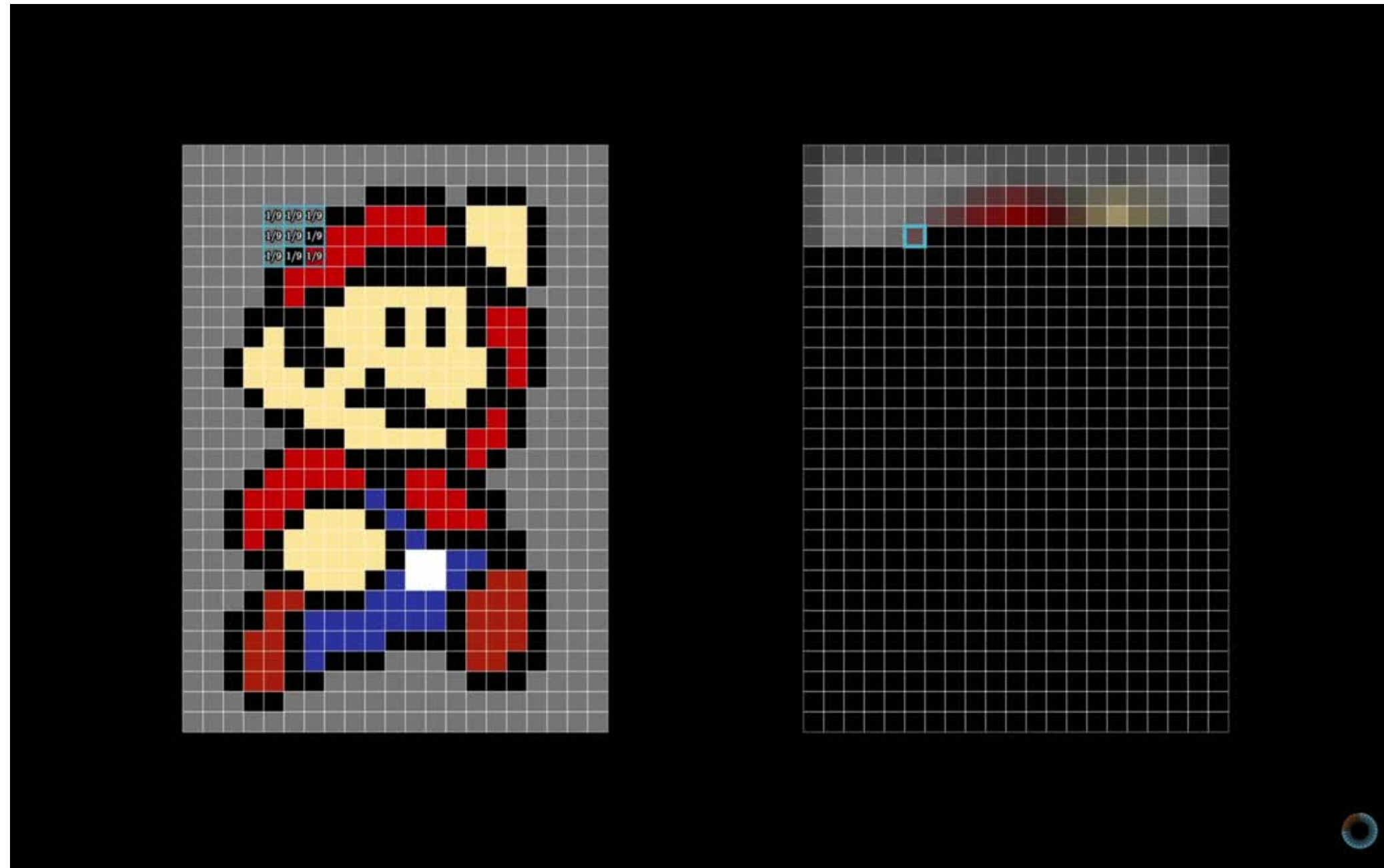
CONVOLUCIÓN: 2D

$$y[n] = \sum_{k=-\infty}^{\infty} x[k]h[n-k]$$



$$(I * K)[i, j] = \sum_{m=-a}^a \sum_{n=-b}^b I[i+m, j+n] \cdot K[m, n]$$

EN ACCIÓN



Fuente: Video de 3Blue1Brown

REDES CONVOLUCIONALES



Suavizado (Blur) Kernel

1/9	1/9	1/9
1/9	1/9	1/9
1/9	1/9	1/9

Comparación de Filtros de Convolución

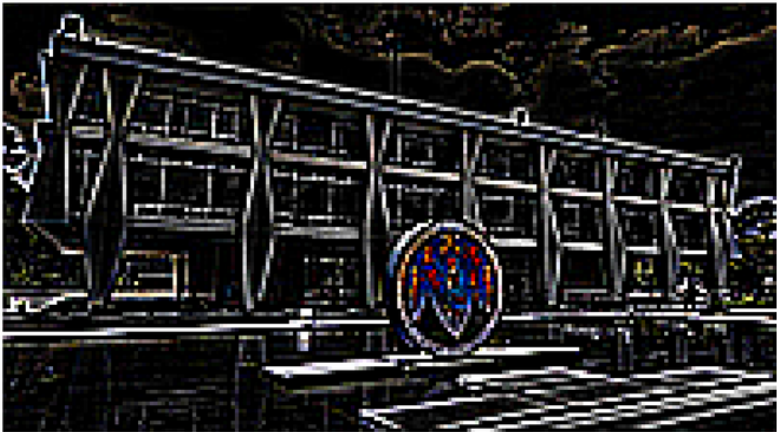
Suavizado (Blur) Result



Detección de Bordes Kernel

-1	-1	-1
-1	8	-1
-1	-1	-1

Detección de Bordes Result



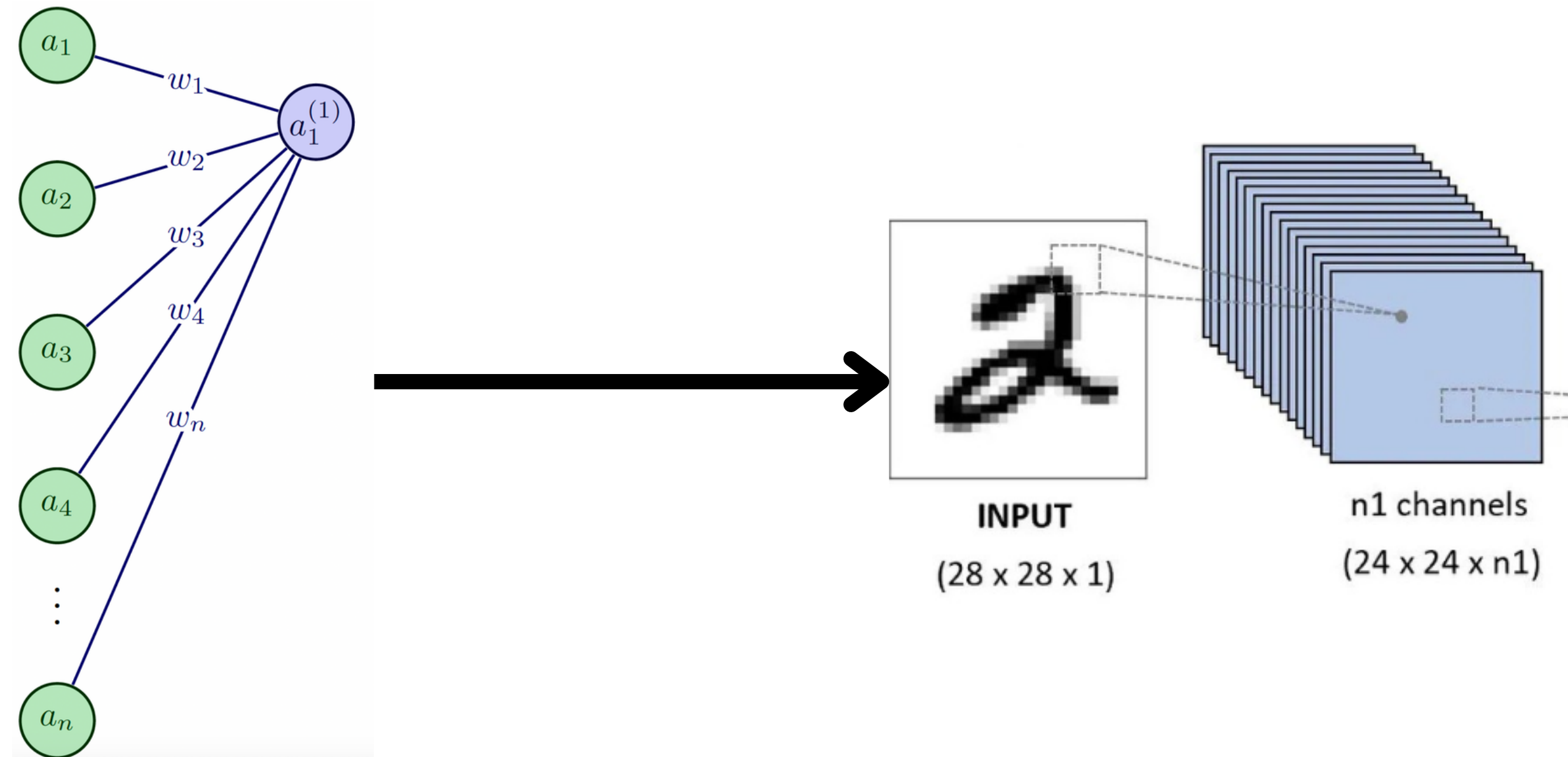
Afilado (Sharpen) Kernel

0	-1	0
-1	5	-1
0	-1	0

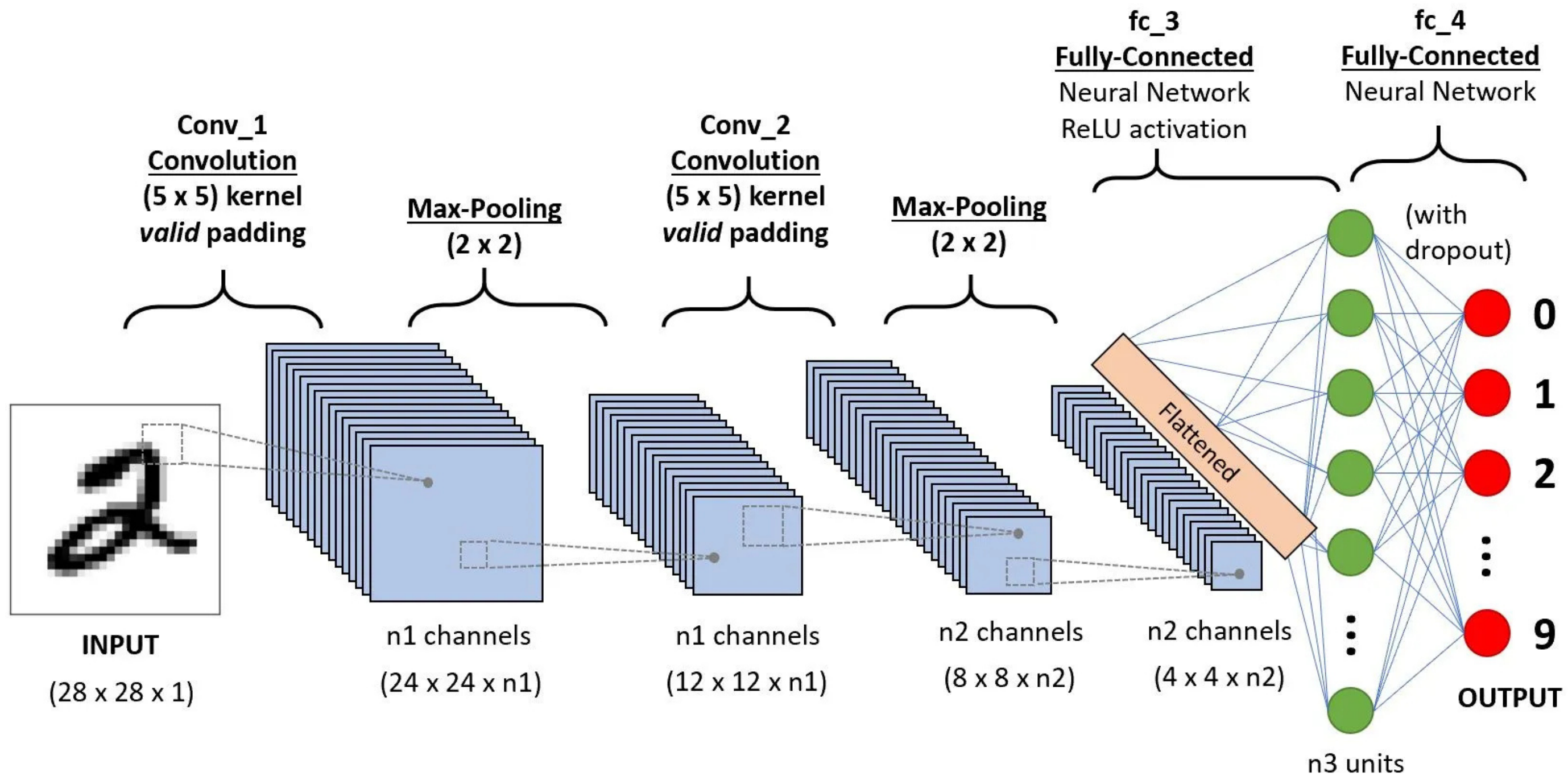
Afilado (Sharpen) Result



¿CÓMO SE IMPLEMENTA EN REDES NEURONALES?



REDES CONVOLUCIONALES



MNIST: UN PROBLEMA CLÁSICO

Un conjunto de números escritos a mano con definición de 28 por 28 pixeles, guardados en csv.



[Fuente: Wikipedia](#)

¿CÓMO RESOLVERLO?

Una pequeña implementación de código.

1. Cargar los datos
2. Arquitectura de la red
3. Compilación
4. Preprocesamiento de datos
5. Entrenamiento
6. Evaluación de resultados
7. Visualizar la red (opcional)

Cargar los datos

```
1 from keras.datasets import mnist
2 (train_images, train_labels), (test_images, test_labels) = mnist.load_data()
```

Arquitectura de la red

```
from keras import models
```

```
from keras import layers
```

```
network = models.Sequential()
```

```
network.add(layers.Dense(512, activation='relu', input_shape=(28 * 28, )))
```

```
network.add(layers.Dense(10, activation='softmax'))
```

Compilación

```
network.compile(optimizer='rmsprop',  
                loss='categorical_crossentropy',  
                metrics = ['accuracy'])
```

Preprocesamiento de datos

```
# Preprocessing training data
train_images = train_images.reshape((60000, 28*28))
train_images = train_images.astype('float32') / 255

# Preprocessing test data
test_images = test_images.reshape((10000, 28 * 28))
test_images = test_images.astype('float32') / 225
```

Entrenamiento

```
from keras.utils import to_categorical
```

```
train_labels = to_categorical(train_labels)
```

```
test_labels = to_categorical(test_labels)
```

Evaluación de resultados

```
# Calculate Test loss and Test Accuracy
test_loss, test_acc = network.evaluate(test_images, test_labels)

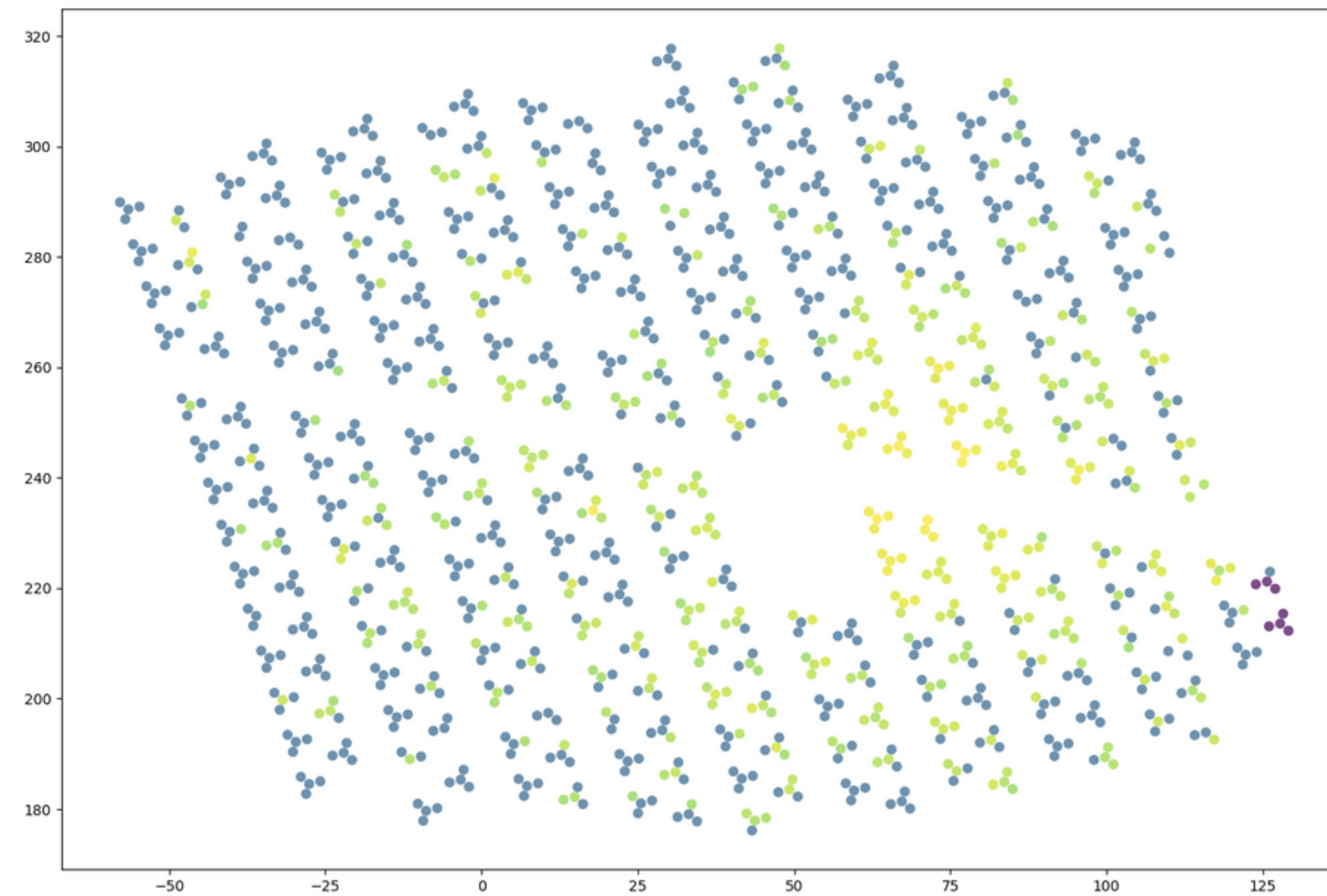
# Print Test loss and Test Accuracy
print(f"Test Loss: {test_loss}\nTest Accuracy : {test_acc * 100} %")
```



HAWC

- Ubicado a 4100 m s.n.m
- En Sierra Negra, Puebla
- 300 tanques del arreglo principal,.
 - Cada uno con 4 PMTs
- Operando desde 2015
- Arreglo exterior con 345 tanques
 - Cada uno con un PMT
- Operando desde el 2018

HAWC: UNA REPRESENTACION

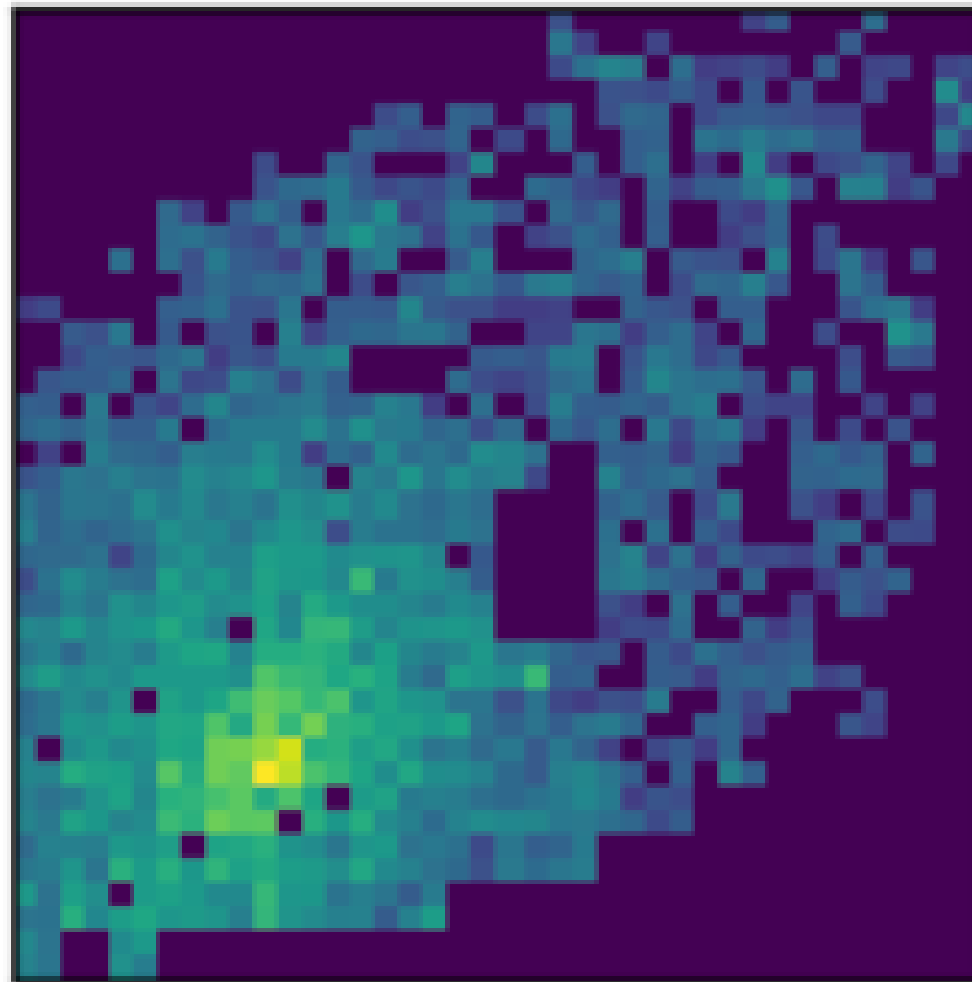


Fuente: [github de arceline](#)

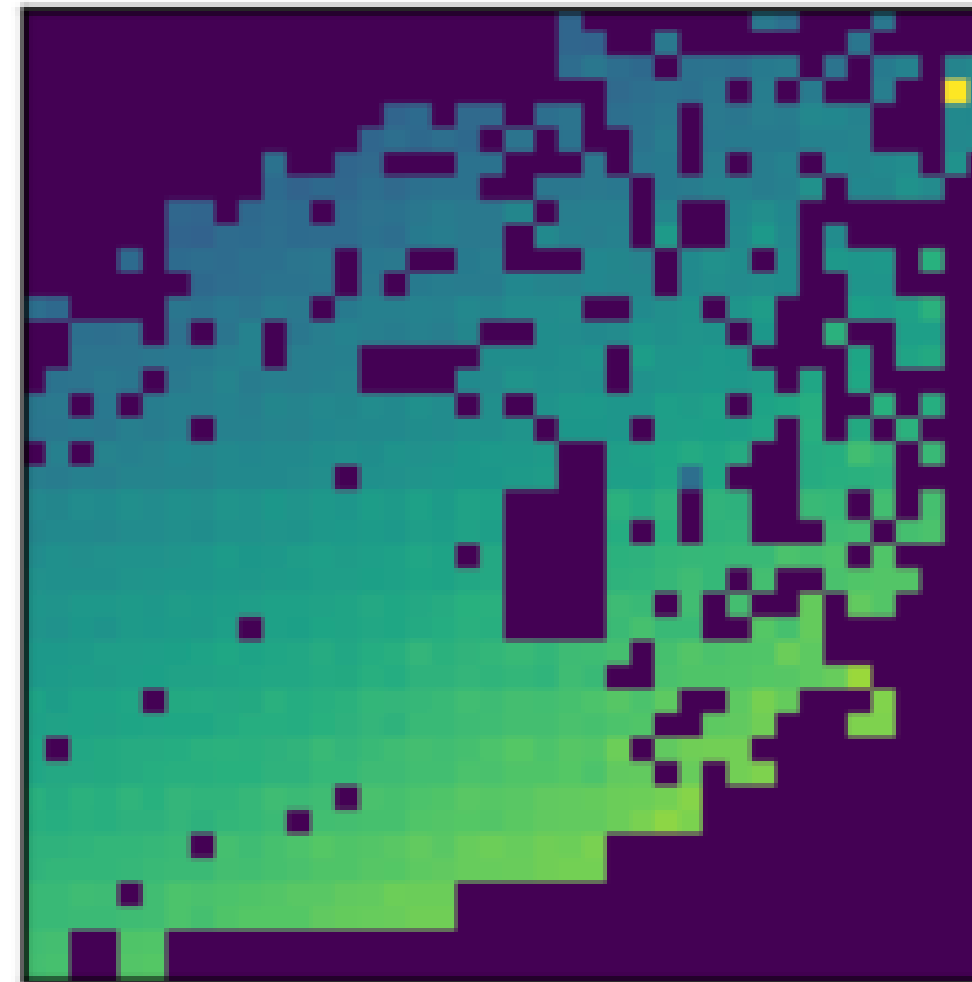
HAWC COMO IMAGEN

Carga y tiempo

event 9, charge



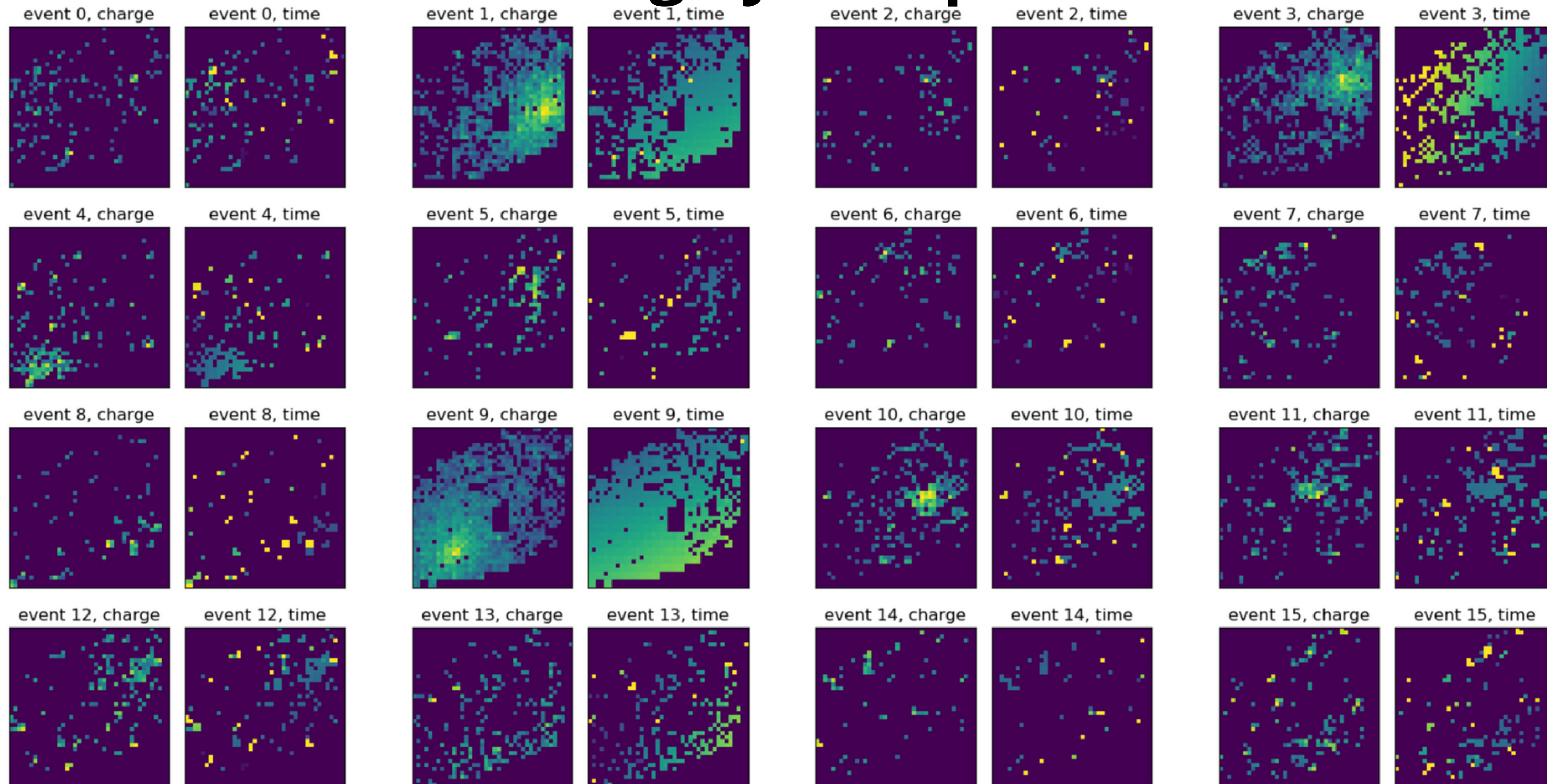
event 9, time



Fuente: [github de arceline](#)

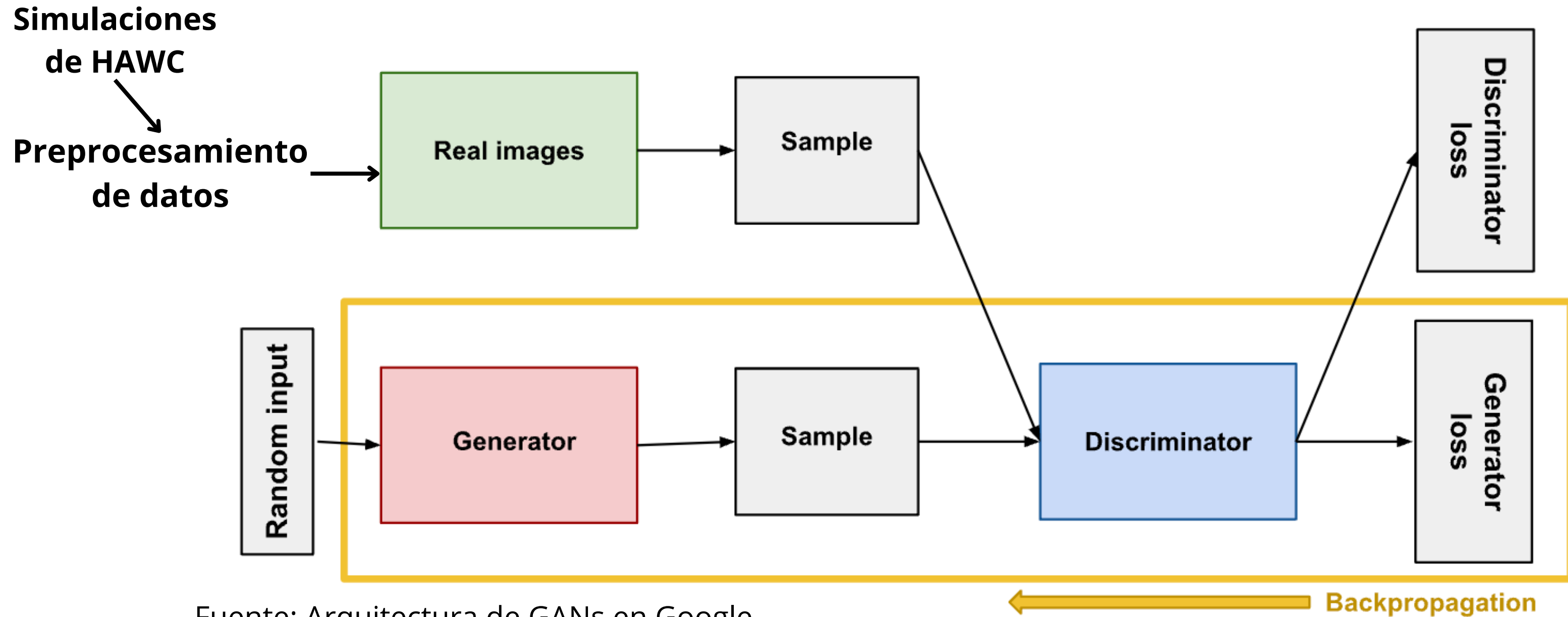
HAWC COMO IMAGEN

Carga y tiempo



Fuente: [github de arceline](https://github.com/arceline)

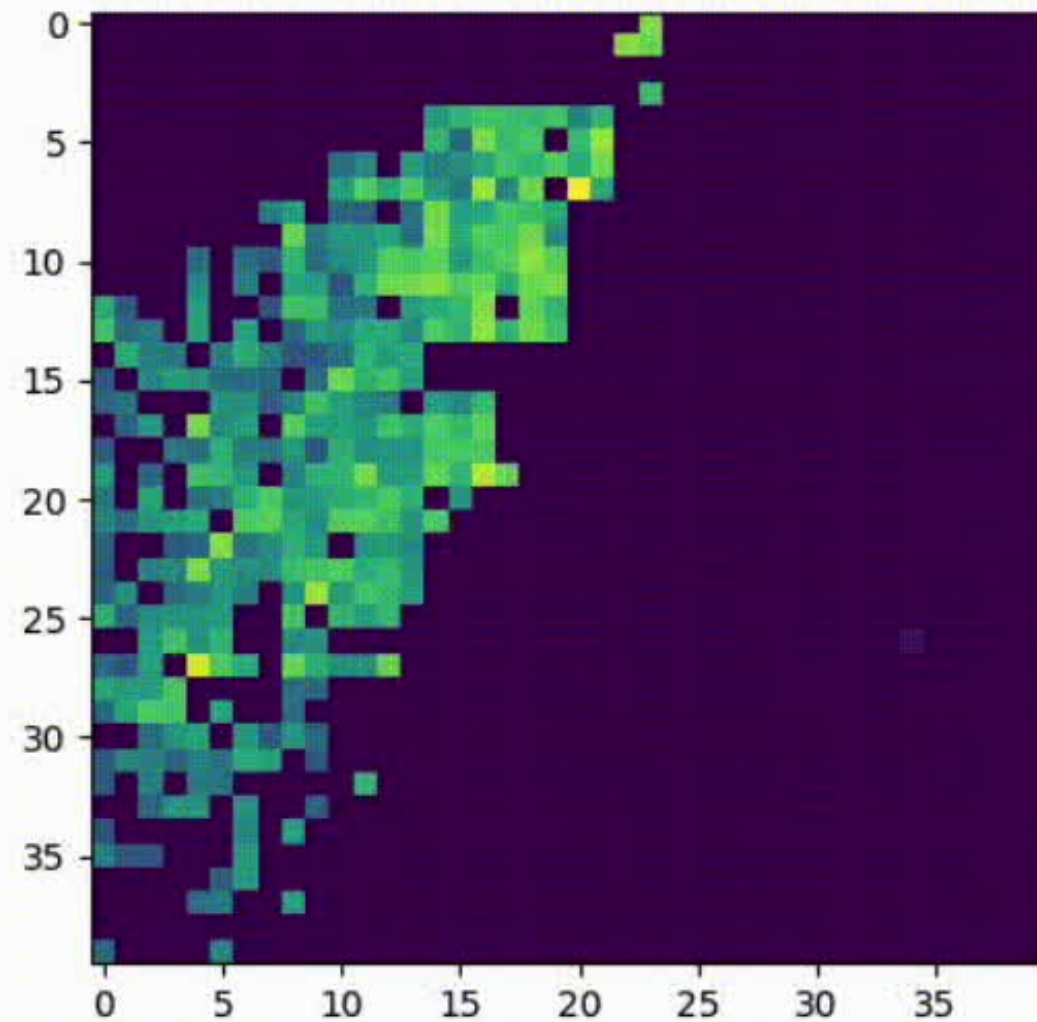
REDES GENERATIVAS ADVERSARIALES



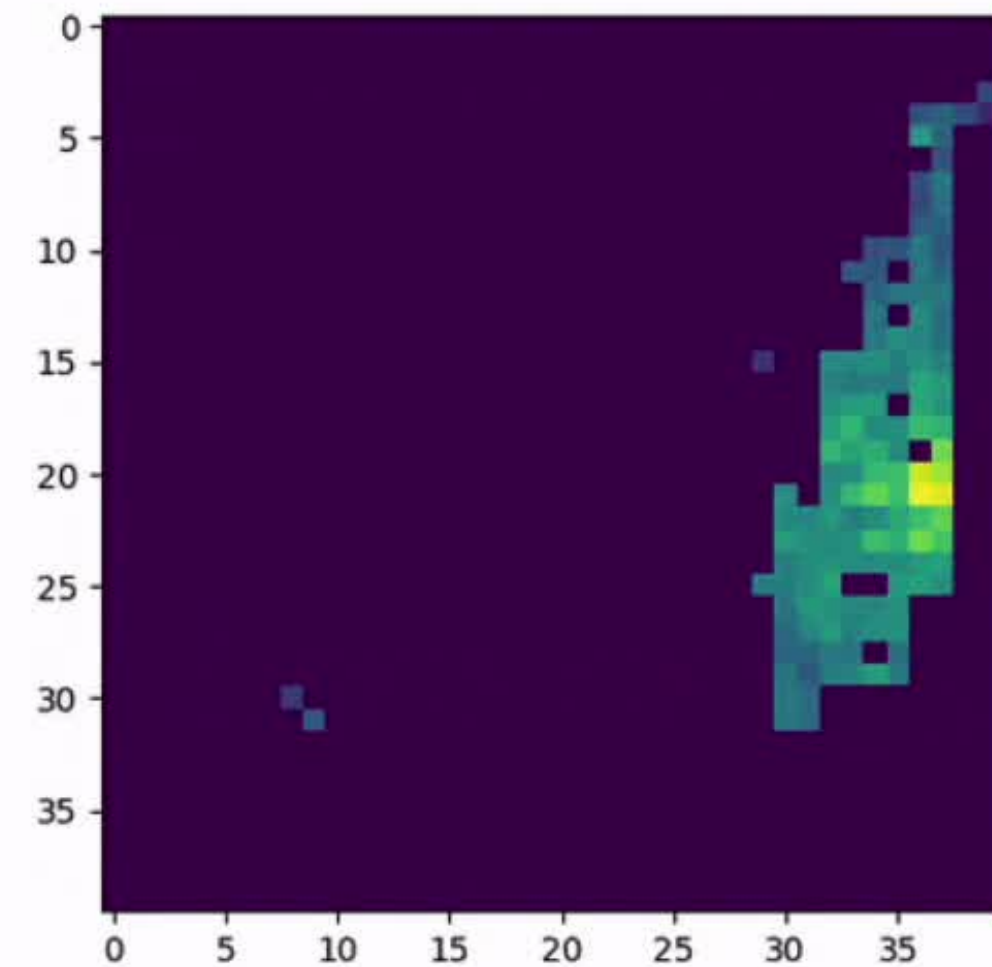
Fuente: Arquitectura de GANs en Google

RESULTADOS

Physics Simulation Data



Generated by Neural Network



Fuente: [github de arceline](#)