import numpy as np

# Données d'entraînement

data = np.array([

 [1, 1, 1],

 [-1, 1, -1],

 [-1, -1, -1],

 [1, -1, -1]

])

# Initialisation des paramètres

w1, w2, b = -0.2, 0.1, -0.2 # Poids et biais

eta = 0.1 # Taux d'apprentissage

# Fonction d'activation (Heaviside)

def activation(x):

 return 1 if x >= 0 else -1

# Entraînement du perceptron

epochs = 10 # Nombre maximal d'itérations

for epoch in range(epochs):

 error\_count = 0

 for e1, e2, d in data:

 # Calcul de la sortie du perceptron

 y = activation(w1 \* e1 + w2 \* e2 + b)

 # Mise à jour des poids si erreur

 if y != d:

 error\_count += 1

 w1 += eta \* (d - y) \* e1

 w2 += eta \* (d - y) \* e2

 b += eta \* (d - y)

 # Arrêt si plus d'erreur

 if error\_count == 0:

 break

# Résultats

print(f"Poids finaux: w1 = {w1}, w2 = {w2}, b = {b}")