

**UNIVERSIDAD AUTONOMA DE OCCIDENTE**  
**FACULTAD DE INGENIERIA.**  
**REDES NEURONALES Y DEEP LEARNING**  
**PROFESORES: JESUS ALFONSO LOPEZ – ANDRES FELIPE ESCOBAR –**  
**JOSE LUIS PANIAGUA**

**Trabajando con el Gradiente Descendente**

El gradiente descendente es la técnica en la cual se basan la mayoría de algoritmos de entrenamiento u optimizadores que se usan para entrenar redes neuronales profundas.

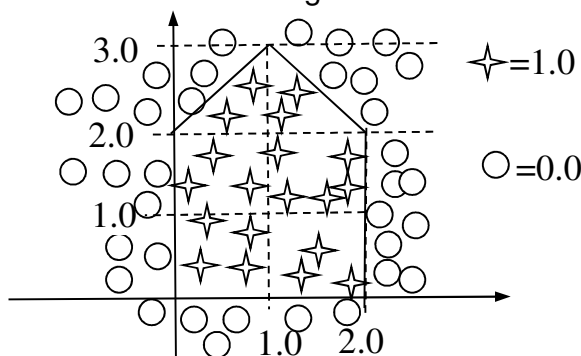
Los siguientes ejercicios le permitirán apropiarse de los conceptos básicos necesarios para entender el efecto de aplicar gradiente descendente a redes monocapa y superficiales.

**Actividades**

1. Realice manualmente (sin usar ninguna herramienta de simulación) la actualización de pesos de una red Adaline que resuelva el problema de la función lógica OR (ver tabla). Pruebe al menos una vez cada una de las combinaciones de entrada, luego de hacer esto, calcule el valor de la de la pérdida (loss) con que quedó la red entrenada. Use como valores de pesos iniciales  $w_1=0.5$   $w_2=-0.5$   $b=-1.5$ . Seleccione un valor del parámetro de aprendizaje entre 0.5 y 1.0

X1	X2	D
0	0	0
0	1	1
1	0	1
1	1	1

2. Encuentre una red neuronal multicapa superficial compuesta de neuronas con función de activación tipo escalón que permita realizar la separación de los patrones mostrados en la figura.



3. Resuelva el problema de la función lógica OR usando una red Adaline y una red multicapa superficial en tensorflow-keras. Pruebe al menos tres valores del parámetro de aprendizaje. Use tensorboard para visualizar la evolución de la pérdida (loss) y los grafos de las redes neuronales creadas. ¿Qué puede concluir al variar el valor del parámetro de aprendizaje?
4. Resuelva el problema de la función lógica XOR (ver la siguiente tabla) usando una red Adaline y una red multicapa superficial en tensorflow-keras. Use tensorboard para visualizar la evolución de la pérdida (loss) y los grafos de las redes neuronales creadas.

X1	X2	D
0	0	0
0	1	1
1	0	1
1	1	0

5. Use el playground de tensorflow (<https://playground.tensorflow.org/>) y verifique con los data set “circle” y “spiral” el efecto de ir aumentando las capas en una red neuronal. Comience con una red de una sola capa de procesamiento y vaya agregando capas ocultas y/o neuronas hasta resolver el problema convenientemente. Documente el proceso realizado
6. Resuelva en tensorflow-keras el problema de clasificación de datos circulares y de espiral trabajados en playground de tensorflow en el punto anterior. (<https://playground.tensorflow.org/>)  
Use tensorboard para visualizar la evolución de la pérdida (loss) y los grafos de las redes neuronales creadas.
7. Realice una comparación al aplicar gradiente descendente convencional y gradiente descendente con momentum usando simulaciones que se pueden realizar usando el siguiente enlace (<https://www.deeplearning.ai/ai-notes/optimization/>)  
Utilice por lo menos dos paisajes de costo (*cost landscape*) y modifique los parámetros de entrenamiento. Documente el proceso realizado
8. Realice una breve investigación sobre qué es la diferenciación automática y muestre un ejemplo de cómo se puede implementar para entrenar redes neuronales artificiales  
Nota: Se sugiere buscar el término en inglés “automatic differentiation”