

## Tarea 2

1) Se realizó una implementación de perceptron simple para clasificar datos en dos clases, usando el lenguaje de programación Python, guardando dicha implementación en el archivo perceptron.py. Dicha implementación consta de una clase perceptron y dos funciones: una para clasificar elementos, usando la función signo con salidas 1 y -1; y una segunda función para entrenar el perceptron con un conjunto de datos provenientes de un archivo csv. El perceptron inicia con todos sus pesos en cero, y para realizar el entrenamiento se puede escoger entre pasar las instancias en el orden que se extraen del archivo, o en un orden aleatorio.

Usando dicha implementación para entrenar y clasificar con los datos encontrados en el sitio web <http://archive.ics.uci.edu/ml/machine-learning-databases/iris/bezdekIris.data>, separando la Iris Setosa y asignándole el valor deseado 1, y juntando las clases Iris virginica e Iris versicolor y asignándoles el valor deseado -1. Se obtuvieron los siguientes resultados:

Entrenando la red pasando los estímulos en orden aleatorio, y usando tasas de aprendizaje de 0.1, 0.01 y 0.0000000005, el perceptron logró clasificar todas las muestras luego de 2 épocas. Usando las mismas tasas de aprendizaje pero pasando los estímulos en el orden en que son extraídos del archivo, se logró clasificar todas las muestras correctamente luego de 4 épocas.

Con estos resultados podemos concluir que para la muestra dada, la tasa de aprendizaje del perceptron no afecta los resultados, mientras que el orden en que se usan los estímulos para entrenar el perceptron si es de relevancia, logrando una convergencia más rápida cuando se usan muestras aleatoriamente. Además se observa un patrón en los pesos finales obtenidos, en el que el sesgo y los pesos de los dos primeros parámetros son positivos, y el peso de los dos últimos parámetros son negativos.

2) Se realizó una implementación de perceptron múltiple para clasificación de datos en 2 o más clases. La implementación fue realizada igualmente usando Python y se encuentra en el archivo perceptronCompetitivo.py. Cada uno de los perceptrones inicializa los pesos en cero.

Se entrenó la red con los datos encontrados en el archivo 4D.xls, que consta de 4 clases y 4 parámetros, y se obtuvieron los siguientes resultados:

Entrenando la red pasando los estímulos en orden aleatorio, y usando tasas de aprendizaje de 0.1, 0.01 y 0.0000000005, en diversas corridas del programa la red logró clasificar todas las muestras en una cantidad de épocas entre 2 y 6, sin importar la tasa de aprendizaje. Usando las mismas tasas de aprendizaje pero pasando los estímulos en el

orden en que son extraídos del archivo, se logró clasificar todas las muestras correctamente luego de 2 épocas, igualmente sin importar la tasa de aprendizaje.

Usando los datos proporcionados, se observa que la tasa de aprendizaje no afecta la convergencia del algoritmo, mientras que el orden en que se pasan los estímulos sí, aunque en este ejercicio ambas muestras son aleatorias ya que el archivo proporciona datos sin un orden específico en las clases. Para el vector de pesos de la primera neurona no se observa un patrón claro, en el segundo vector se observa que todos sus valores a excepción del sesgo son negativos, el tercer vector posee sesgo negativo y el resto de sus valores positivos, y en el cuarto vector se observa que las primeras tres componentes son positivas y las últimas dos negativas.

3) Debemos demostrar que las clases de la función lógica O-exclusivo no son linealmente separables. Para ellos, observamos los valores de los parámetros y resultados de las clases para dicha función, usando la tabla de verdad:

Para efectos de la demostración usaremos 1 para Verdad, y -1 para Falso

Tabla de verdad para la función lógica O-exclusivo

| X(1) | X(2) | Clase |
|------|------|-------|
| -1   | -1   | -1    |
| 1    | -1   | 1     |
| -1   | 1    | 1     |
| 1    | 1    | -1    |

Para la demostración usaremos la técnica de reducción al absurdo. Suponemos que las clases SI son linealmente separables, entonces existe un vector W tal que

$$WtX > 0 \text{ si } X \text{ pertenece a la clase } 1$$

$$WtX < 0 \text{ si } X \text{ pertenece a la clase } -1$$

Usando esta regla, procedemos a clasificar las muestras de la función lógica O-exclusivo:

Sea  $W = (a, w_1, w_2)$ .

$$1) \text{ Para } X = (1, -1, -1), (a * 1 + w_1 * (-1) + w_2 * (-1)) = (a - w_1 - w_2) < 0$$

$$2) \text{ Para } X = (1, 1, -1), (a * 1 + w_1 * 1 + w_2 * (-1)) = (a + w_1 - w_2) > 0$$

$$3) \text{ Para } X = (1, -1, 1), (a * 1 + w_1 * (-1) + w_2 * 1) = (a - w_1 + w_2) > 0$$

$$4) \text{ Para } X = (1, 1, 1), (a * 1 + w_1 * 1 + w_2 * 1) = (a + w_1 + w_2) < 0$$

Resolviendo:

$$1) -w_1 - w_2 < -a$$

$$2) w_1 - w_2 > -a$$

$$3) -w_1 + w_2 > -a$$

$$4) (w_1 + w_2) < -a$$

Estas ecuaciones deben ser ciertas para cualesquiera valores de  $w_1$ ,  $w_2$  y  $a$ , pero se puede encontrar un contraejemplo:

Para  $a = 4$ ,  $w_1 = 2$ ,  $w_2 = 3$

1)  $-2 - 3 < -4 \Rightarrow -5 < -4$  (correcto)

2)  $2 - 3 > -4 \Rightarrow -1 > -4$  (correcto)

3)  $-2 + 3 < -4 \Rightarrow 1 > -4$  (correcto)

4)  $2 + 3 < -4 \Rightarrow 5 < -4$  (incorrecto)

Encontrando así una contradicción, por lo que las clases no son linealmente separables.