

Inteligencia Artificial

*Reporte de Práctica No. 2 – Perceptrón Multicapa*

**Docente:** Ernesto Alejandro Lima Solana

**Alumno:** Carriola Monroy Angel Fernando

**NC:** 13012148

**Fecha:** 24 de abril de 2017

# PREGUNTAS

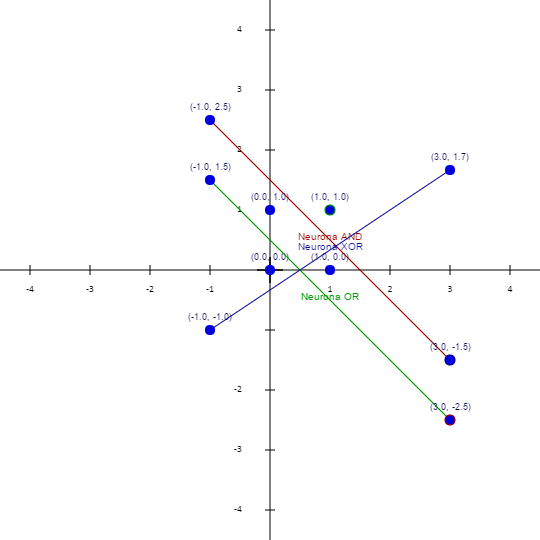
1. **Entrene la neurona 1 por separado, entrene la neurona 2 por separado. Introduzca los valores de salida de las neuronas 1 y 2 a la neurona 3.**

**Nota: la neurona 1 es OR y la neurona 2 es AND.**

* 1. **Anote los pesos óptimos de aprendizaje en las 3 neuronas, así como el umbral óptimo.**

|  |  |  |  |
| --- | --- | --- | --- |
|  | ω1 | ω2 | θ |
| Neurona 1 | 1 | 1 | 0.5 |
| Neurona 2 | 1 | 1 | 1.5 |
| Neurona 3 | 1 | -1.5 | 0.5 |

* 1. **Grafique las rectas pertenecientes a las neuronas 1 y 2.**



(Recta Roja: AND, recta Verde: OR)

* 1. **¿Qué sucede si el umbral inicial en las neuronas 1 y 2 se inicializa con cero?**

A ambas neuronas les toma más tiempo aprender, y, además, las rectas generadas tienden a cruzarse entre sí, obteniendo un error de desbalanceo al resultado esperado, es decir, las rectas no logran ser completamente paralelas entre sí. Sin embargo, se logra aún clasificar las clases cero y uno.

* 1. **¿Qué sucede si el umbral inicial en las neuronas 1 y 2 se inicializa con uno?**

A ambas neuronas les toma menos tiempo aprender, menos épocas, y, además, las rectas generadas tienden a cruzarse bruscamente, obteniendo un gran error al resultado esperado pues además de no ser paralelas entre sí, tampoco se logra clasificar correctamente en la recta generada por OR.

* 1. **¿Por qué se puede afirmar que hay infinitas combinaciones de los pesos que hacen el error nulo para la función OR, y a su vez hay una única combinación de los pesos para que los pesos sean nulos en el AND, y por qué no hay ninguna combinación de los pesos que haga nulo el error en el XOR?**

Para justificarlo, primero se muestran las tablas de verdad de cada compuerta.

|  |  |  |
| --- | --- | --- |
| **AND** | | |
| x1 | x2 | **t** |
| 0 | 0 | **0** |
| 0 | 1 | **0** |
| 1 | 0 | **0** |
| 1 | 1 | **1** |

|  |  |  |
| --- | --- | --- |
| **OR** | | |
| x1 | x2 | **t** |
| 0 | 0 | **0** |
| 0 | 1 | **1** |
| 1 | 0 | **1** |
| 1 | 1 | **1** |

|  |  |  |
| --- | --- | --- |
| **XOR** | | |
| x1 | x2 | **t** |
| 0 | 0 | **0** |
| 0 | 1 | **1** |
| 1 | 0 | **1** |
| 1 | 1 | **0** |

Ahora se muestran las cuatro desigualdades que deben cumplir los pesos y el umbra para que puedan existir combinaciones que hagan el error nulo dependiendo la tabla de verdad.

**XOR**

**OR**

**AND**

Como puede verse en AND, los pesos siempre deben ser menores al umbral, pero la suma de estos mayor o igual este. Por ello no es posible asignar cualquier tipo de peso, pues hay que poner una única combinación para cumplir con estas cuatro desigualdades.

En el caso de OR, puede notarse que los pesos siempre deben ser mayores o iguales al umbral ya sea de manera independiente o sumándolos, por lo que es evidente que las combinaciones pueden ser infinitas debido a que los pesos pueden ser tan grandes como se desee.

Finalmente, para XOR, puede notarse claramente cómo se contradicen estas desigualdades, ya que la desigualdad dos y tres mencionan que los pesos deben ser mayores o iguales al umbral, pero la cuarta desigualdad dice que la suma de estos siempre debe ser menor a este, claramente es imposible cumplir todas estas desigualdades a la vez, por tanto, no existe ningún tipo de combinación en los pesos para que el error sea nulo.

# CONCLUSIÓN

En esta práctica aprendí cómo es posible solucionar el problema de la función XOR al no ser linealmente separable con un perceptrón simple. Dado que las clases 0 y 1 en una hiperplano no podían ser separadas con una simple recta, en esta práctica se planteó la solución usando un perceptrón de multicapa donde por medio de una neurona AND y una OR (capa oculta), es posible clasificar los puntos usando dos rectas resultantes de su entrenamiento para finalmente la salida de ambas entregarlas a las entradas de una tercera neurona XOR.

También se pudo entender que las cuatro desigualdades para ajustar los pesos y el umbral de la neurona varían dependiendo la compuerta a utilizar y que en el caso de XOR es imposible lograr que el error sea nulo ya que existe una contradicción que hace que los pesos jamás puedan ser ajustas.

Finalmente fue interesante reutilizar el programa para el perceptrón simple con la intención de elaborar el perceptrón multicapa utilizando el mismo algoritmo de aprendizaje en las tres neuronas, pero cambiando las entradas y compuerta lógica (AND, OR, XOR). Además, se pudo apreciar gráficamente la clasificación de los puntos en el plano usando dos rectas generadas por la neurona AND y OR.