

# Redes Neuronales Monocapa

García Prado, Sergio  
sergio@garciparedes.me

Fernández Angulo, Óscar  
oscar.fernandez.angulo@alumnos.uva.es

10 de mayo de 2017

## Resumen

[TODO]

## 1. INTRODUCCIÓN

[TODO]

### 1.1. PERCEPTRÓN SIMPLE

[TODO]

### 1.2. ADALINE

[TODO]

## 2. PERCEPTRÓN SIMPLE SOBRE CONJUNTO DE DATOS SENCILLO

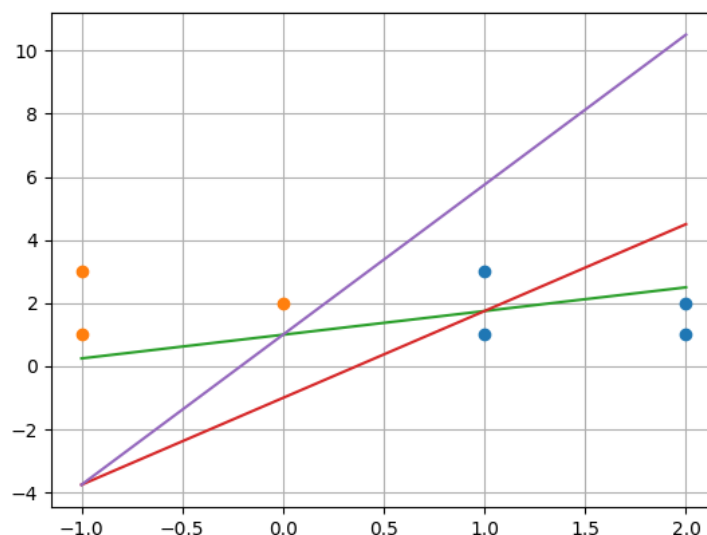
En esta sección se describe el experimento realizado utilizando la estrategia del *perceptrón simple*. El conjunto de datos utilizado se muestra en la tabla 1. Este conjunto de datos posee la característica de ser linealmente separable, tal y como se verá a continuación. Está formado por **8 instancias** formadas por **2 atributos** de carácter numérico. La clase de destino también es de carácter numérico, formada por el conjunto de valores  $D = \{1, -1\}$ . Puesto que este conjunto está formado por dos categorías, entonces se puede utilizar un único clasificador binario (perceptrón simple en este caso) para clasificar las instancias.

En este caso, debido al reducido tamaño del conjunto de datos, se ha realizado un experimento basado en el error de resustitución. Por lo tanto, se ha utilizado el 100 % de las instancias, tando para entrenamiento como para test.

Simple Dataset		
$x_1$	$x_2$	$d(x)$
1	1	1
2	1	1
2	2	1
1	3	1
-1	1	-1
0	2	-1
-1	3	-1

**Tabla 1:** *Conjunto de datos Simple*

Puesto que las el conjunto de datos es linealmente separable tal y como se ha indicado anteriormente (se puede apreciar en la figura 1) y la metodología experimental seguida es se apoya en el error de resubstitución. La tasa de error obtenida es del 0,0 % tal y como se muestra en la tabla 2.



**Figura 1:** Evolución de los pesos conforme aumenta el número de iteraciones

La figura 1 muestra la evolución de la recta que separa las dos clases de datos conforme aumentan las iteraciones. La línea en color **verde** se corresponde con la **iteración inicial**, la línea en color **verde** se corresponde con la **primera iteración** y, por último, la línea **púrpura** se corresponde con la **segunda iteración**. Nótese que en este punto las dos clases están separadas por dicha recta, por lo que el algoritmo ya está en condiciones de terminar.

Perceptrón Simple - Simple Dataset	
Error de Resubstitucion	0,0 %

**Tabla 2:** Resultados del experimento sobre el conjunto de datos Simple

### 3. ADALINE SOBRE MACHINE DATASET

En esta sección se describe el experimento realizado sobre el conjunto de datos **Computer Hardware**[UCI] y el algoritmo **Adaline**. Puesto que el algoritmo de aprendizaje realiza una regresión sobre los datos, para calcular la tasa de aciertos se ha utilizado como unidad de medida un **error relativo del 10 %** sobre el valor real. Los parámetros utilizados para el ADALINE han sido un ratio de aprendizaje  $\alpha = 0,1$  y 5000 épocas.

La metodología experimental que se ha seguido ha sido un **HoldOut** con particionamiento de los datos de manera que  $\frac{2}{3}$  son utilizados en la fase de entrenamiento y  $\frac{1}{3}$  en la de test. Los resultados obtenidos tras dicha experimentación se recogen en la tabla 3. Tal y como se puede apreciar, la tasa de error obtenida es muy elevada. Esto puede ser debido a la existencia de instancias muy diferentes unas de otras, lo cual penaliza drásticamente los resultados.

Adaline — Computer Hardware Dataset	
Error HoldOut	71,429 %

**Tabla 3:** *[TODO]*

Tras realizar un experimento que utilizara todo el conjunto de datos (tanto para entrenamiento como para test) el error obtenido se redujo a 0,0 %. Esta tarea se ha llevado a cabo para comprobar la corrección de la implementación utilizada. Sin embargo, nótese que esta circunstancia no es significativa para comprobar la validez del clasificador ADALINE sobre el conjunto de datos Computer Hardware puesto que en un entorno real esta situación no es posible.

## REFERENCIAS

- [CCAG17] Teodoro Calonge Cano and Carlos Javier Alonso González. Técnicas de Aprendizaje Automático, 2016/17.
- [UCI] UCI Machine Learning Repository. Computer Hardware Data Set. <http://archive.ics.uci.edu/ml/datasets/Image+Segmentation>.