

# Tarea 2

María Victoria Jorge  
11-10495

October 3, 2016

## 1 Clasificadores Lineales

En esta sección se presentarán los experimentos realizados para construir clasificadores lineales mediante aprendizaje con reforzamiento, un perceptrón de una capa y un adaline.

Para medir la calidad de todos los modelos se utilizará la técnica de validación cruzada. Además, se utilizará el porcentaje de datos bien clasificados como métrica de *performance*. Por último, para asegurarnos que los modelos no están aprendiendo a clasificar basándose en el orden en el que reciben los datos, en cada corrida estos se desordenarán aleatoriamente.

Como los datos presentados en el archivo lincloud3 tienen solo dos dimensiones, pueden graficarse e identificar fácilmente si son linealmente separables, para asegurarnos que los modelos utilizados a continuación serán capaces de clasificarlos. En la figura 1 se tienen todos los datos identificados según la clase a la que pertenecen, y se puede observar que es posible trazar una línea entre ambas clases para separarlas.

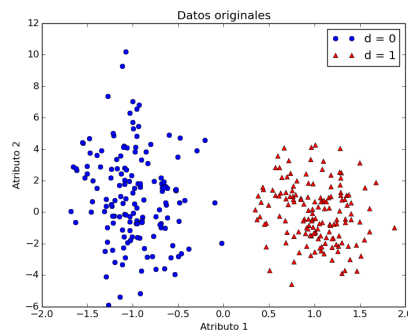


Figure 1: Datos originales

### 1.1 Aprendizaje con reforzamiento

Para estas pruebas se inicializaron los pesos de las dos neuronas en valores aleatorios entre -0.5 y 0.5, y se usaron esos mismos valores en todas las corridas.

Como condición de parada para el entrenamiento se tomó que no existiera error en la clasificación de los datos o que se superara un número máximo de épocas (1000 épocas).

En la Tabla 1 se encuentran los resultados obtenidos de la ejecución de la validación cruzada 5-fold usando la red basada en aprendizaje con reforzamiento. Se observa que para las tres tasas de aprendizaje utilizadas se obtuvieron resultados parecidos, clasificando correctamente la mayoría de los datos. El número de épocas indicado es el promedio de épocas utilizadas durante las 5 iteraciones de la validación cruzada.

Total datos	$\alpha$	% Correctos	% Incorrectos	Épocas
300	0.1	99.6667	0.3333	4.8
300	0.01	99.6667	0.3333	2
300	0.001	99.3333	0.6667	4.2

Table 1: Exactitud aprendizaje con reforzamiento

## 1.2 Perceptrón

Para estas pruebas se inicializaron los pesos en valores aleatorios entre -0.5 y 0.5, y se usaron esos mismos valores en todas las corridas. Se utilizó una sola neurona y la función de activación  $\text{sgn}$  o umbral. Como condición de parada para el entrenamiento se tomó que no existiera error en la clasificación de los datos o que se superara un número máximo de épocas (1000 épocas). Además se cambió la respuesta deseada de los datos cuyo valor era 0, por -1, para que coincidiera con el rango de valores que devuelve la función  $\text{sgn}$ .

En la Tabla 2 se encuentran los resultados obtenidos luego de ejecutar validación cruzada con 5-fold para diferentes tasas de aprendizaje en una única neurona. Por cada iteración se utilizaron 240 datos para entrenar y 60 para probar el modelo. Luego de 5 iteraciones se probó en todos los experimentos con los 300 datos. El número de épocas indicado para cada tasa de aprendizaje es el promedio de los 5 entrenamientos realizados por tasa, con diferentes datos para entrenar en cada una.

Con estos resultados puede observarse que al disminuir la tasa de aprendizaje el perceptrón necesita más épocas para cumplir con la cota de clasificar correctamente todos los datos de entrenamiento. Sin embargo, parece que al aumentar la cantidad de épocas para estos datos, el perceptrón tiene más problemas para clasificar los datos nuevos.

Total datos	$\alpha$	% Correctos	% Incorrectos	Épocas
300	0.1	99.6666	0.3334	3
300	0.01	99.6666	0.3334	3.4
300	0.001	99.3333	0.6667	15.2

Table 2: Exactitud del perceptrón

Esto puede apreciarse de forma más clara en las gráficas de la figura 2, donde se muestra para cada tasa de aprendizaje utilizada el peor hiperplano generado de todas las iteraciones de la validación cruzada. Podemos observar que en las tres figuras el hiperplano está muy cerca de las dos fronteras, lo que puede originar errores de clasificación cuando esos datos no se encuentran dentro del conjunto de entrenamiento, como sucedió en las gráficas 2b y 2c. Con esto se puede concluir que el modelo tiene problemas para generalizar.

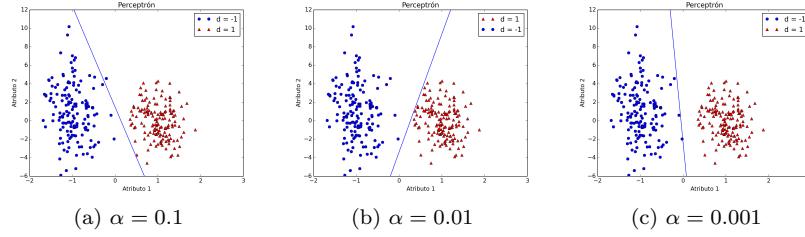


Figure 2: Modelos de perceptrón obtenidos en los tres experimentos

### 1.3 Adaline

Para estas pruebas se inicializaron los pesos de la neurona en valores aleatorios entre -0.5 y 0.5, manteniéndolos fijos durante todas las iteraciones de la validación cruzada de 5-fold. Como condición de parada para el algoritmo de entrenamiento se usó un máximo de 5000 épocas. Además, se modificó el archivo que contiene los datos para que las clases sean -1 y 1, en vez de 0 y 1 como estaban originalmente.

En la Tabla 3 se encuentran los resultados obtenidos para diferentes tasas de aprendizaje. Se puede observar que con tasas de aprendizaje muy pequeñas el Adaline converge mejor a la solución esperada, clasificando correctamente todos los datos.

Total datos	$\alpha$	% Correctos	% Incorrectos	Épocas
300	0.1	98	2	5000
300	0.01	100	0	5000
300	0.001	100	0	5000

Table 3: Exactitud del Adaline

Se realizaron pruebas con otras tasas de aprendizaje pero se presentaron solo las más significativas. El valor más alto con el que se obtuvieron buenos resultados fue  $\alpha = 0.1$ . Con  $\alpha$  más grandes, 5000 épocas no son suficientes para que el Adaline genere porcentajes tan altos de aciertos, debido a que se le dificulta la convergencia.

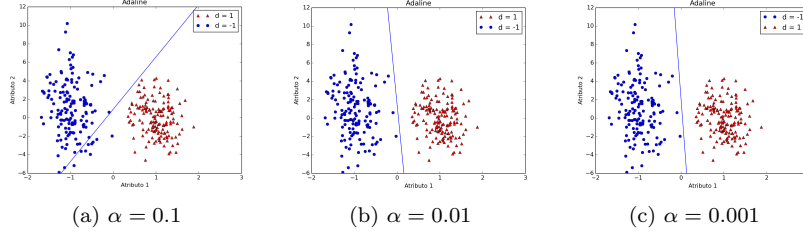


Figure 3: Modelos de Adaline obtenidos en los tres experimentos

## 2 Interpolador

El algoritmo de Adaline puede ser utilizado para interpolación de funciones si se calcula la salida de la siguiente forma  $y = a + b_1x + b_2x^2 + \dots + b_kx^k$  siendo  $k$  el grado del polinomio que se usará para interpolar. Siendo 'a' el sesgo, y cada  $b_i$  los  $w_i$  comunes del Adaline.

Sabiendo esto debemos conocer cómo se comportan los datos que tenemos en p3data para saber el grado del polinomio con el que aproximaremos la función que los originó. Con la figura 4 se puede apreciar que los datos parecen ser originados por una función cúbica, aunque tengan un poco de ruido. Sin embargo, se realizarán pruebas con polinomios de grado 1, 2 y 3 con diferentes tasas de aprendizaje para determinar cuál verdaderamente aproxima mejor estos datos.

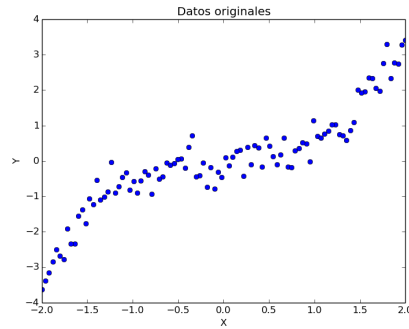


Figure 4: Datos originales a interpolar

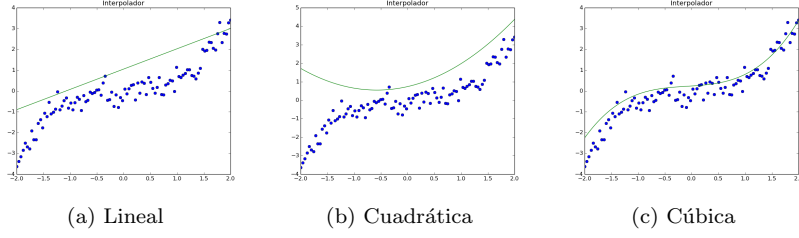


Figure 5: Aproximaciones con  $\alpha = 0.01$

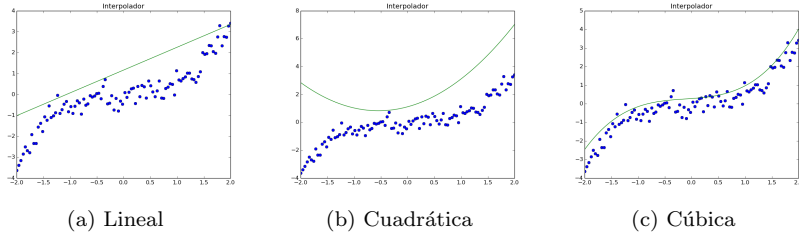


Figure 6: Aproximaciones con  $\alpha = 0.01$

### 3 O-exclusivo

Se demostrará que las clases del O-exclusivo (XOR) no son linealmente separables por reducción al absurdo.

Supongamos entonces que las clases del XOR son linealmente separables. Entonces existe un  $\vec{w}^t = [w_0, w_1, w_2]^t$  que permite generar el hiperplano  $\vec{w}^t \vec{x} = 0$  para un perceptrón de una neurona. Esta es la frontera que divide a las dos clases del XOR. Los datos que estén por debajo de esa frontera pertenecen a la clase 0, y los que están por encima pertenecen a la clase 1.

Por lo tanto se generan las siguientes ecuaciones:

$$0 * w_1 + 0 * w_2 + w_0 \leq 0 \iff w_0 \leq 0 \quad (1)$$

$$0 * w_1 + 1 * w_2 + w_0 > 0 \iff w_0 > -w_2 \quad (2)$$

$$1 * w_1 + 0 * w_2 + w_0 > 0 \iff w_0 > -w_1 \quad (3)$$

$$1 * w_1 + 1 * w_2 + w_0 \leq 0 \iff w_0 \leq -w_1 - w_2 \quad (4)$$

De (1) obtenemos que  $w_0$  es un número no positivo. Usando esta información en (2) y (3) tenemos que  $w_1$  y  $w_2$  son números positivos.

Luego, sabiendo que  $w_1$  es positivo, podemos asegurar que  $-w_2 > -w_2 - w_1$ . Usando esta cota inferior en (2) se sigue cumpliendo que  $w_0 > -w_2 - w_1$ . Esto

entra en contradicción con (4) que se obtuvo directamente del perceptrón. Por lo tanto la suposición que se hizo inicialmente sobre la separabilidad lineal de las clases del XOR es errónea.