

Práctica 1 de Sistemas Conexionistas (2005/2006)

Apartado 1.

Implementar un programa en Matlab que simule el comportamiento de un Adaline que utilice la regla LMS para su aprendizaje. El programa debe permitir:

- Introducir los patrones de entrenamiento
- Introducir la salida deseada
- Almacenar los diferentes pesos del proceso de entrenamiento (pesos iniciales, finales e intermedios)
- Obtener la representación 3D y 2D (curvas de nivel) de la superficie del error que se trata de minimizar en función de los pesos en el intervalo
- Representar la variación del error en función de los vectores de pesos tanto en la superficie de error como en la representación de niveles
- Representar la variación del error respecto del tiempo

Una vez implementado, el Adaline será utilizado en los siguientes casos:

Caso 1. Entrenar el adaline con el siguiente conjunto de patrones de entrenamiento:

Patrones de entrada: (-4,1) (3,2)

Salida Deseada: (5,-2)

El error máximo es 0.0001 y los elementos de procesado no tienen BIAS

- a) Utilizar una velocidad de aprendizaje pequeña y analizar el resultado obtenido
- b) Utilizar una velocidad de aprendizaje alta y analizar el resultado obtenido

Caso 2. Entrenar el adaline (sin bias) con patrones linealmente dependientes (2 dimensiones) y analizar su comportamiento y los resultados obtenidos. Utilizar un error = 0.0001

Apartado 2.

Generar una señal de entrada continua en un determinado intervalo de tiempo. Utilizar una estructura lineal (Adaline) que a partir de vectores de entrada de 3 dimensiones, correspondientes a la señal de entrada muestreada en 3 instantes de tiempo consecutivos, produzca una señal de salida obtenida a partir de una combinación lineal de las 3 muestras de la señal de entrada.

Tras implementar y entrenar el Adaline, deberán tenerse en cuenta los siguientes supuestos:

- a) Las 3 muestras que forman cada patrón de entrada están muy alejadas unas de otras en el tiempo.
- b) Las 3 muestras que forman cada patrón de entrada están muy próximas unas de otras en el tiempo.

Para cada uno de estos casos (a y b):

1. Representar la señal de entrada frente a la señal de salida deseada.
2. Determinar cuáles son los pesos finales de la estructura.
3. Representar el error cometido en función del número de ciclos de entrenamiento.
4. Representar la salida deseada frente a la salida obtenida al finalizar el entrenamiento de la estructura.
5. Comprobar el funcionamiento de la red cuando se le pasa como entrada una señal con ruido.

Apartado 3.

Generar una señal **continua** en el tiempo (en un cierto intervalo), de tal forma que en un cierto subintervalo la señal cambie de forma suave y en otro subintervalo, la señal cambie de forma brusca. Utilizar una estructura lineal para que sea capaz de predecir la señal de salida a partir de valores pasados de la señal actual en un instante de tiempo determinado. Se pide:

- Diseñar la estructura a utilizar (número de entradas y conexiones)
- Generar los patrones de entrenamiento
- Comparar gráficamente la salida obtenida con la salida deseada
- Representar gráficamente el error en función del tiempo
- Realizar el entrenamiento con velocidades de aprendizaje alta y baja y analizar los resultados obtenidos respecto del seguimiento de la señal.