

# Clasificador asociativo de patrones

Héctor Mauricio Zamudio Domínguez

**Abstract**—El uso de memorias asociativas continúa siendo un tema importante de investigación en el tema de la clasificación correcta de patrones. Han surgido propuestas como la Lernmatrix de Steinbuch y en específico para este reporte el Clasificador asociativo de patrones que usa un principio matemático para mejorar la clasificación de patrones binarios.

**Keywords**— Memoria Asociativas, Clasificador asociativo de patrones,

## I. FUNDAMENTOS

El propósito fundamental de una memoria asociativa es recuperar patrones complejos a partir de patrones de entrada que pueden estar alterados con ruido aditivo, sustractivo o combinado. De acuerdo con esta afirmación, una memoria asociativa puede formularse como un sistema de entrada y salida, con un patrón de entrada denotado por  $x$  y el patrón salida denotado por  $y$ . Las memorias asociativas cuentan con dos fases:

Fase de aprendizaje, donde a partir del conjunto fundamental se genera una matriz  $M$  que almacene las asociaciones del conjunto fundamental.

Fase de recuperación. Donde la memoria recibe un conjunto de patrones que clasifica a partir de la matriz  $M$ , creada en la fase anterior. Estos patrones se suelen presentar con ruido. Cuando el vector del patrón salida es de la misma longitud decimos que la memoria es auto asociativa, en caso contrario es una memoria hetero asociativa.

Clasificador asociativo de patrones

Este clasificador, logra solucionar problemas de errores de clasificación para ciertos conjuntos fundamentales, la solución es la translación de ejes, con el nuevo origen situando en el centroide de los vectores que representan a los patrones de entrada. Es importante mencionar que la recuperación es perfecta cuando los patrones están completamente separados por un hiperplano.

## II. PROCEDIMIENTO

El Algoritmo implementado en esta práctica es el siguiente.

Se crea un conjunto fundamental de patrones de dimensión  $n$  con valores reales en sus componentes, los cuales se aglutinan en  $m$  clases diferentes.

Se calcula un vector medio (centroide) del conjunto fundamental. Se toman las coordenadas del vector medio del centro de un nuevo conjunto de ejes coordenados y se realiza una translación de todos los patrones del conjunto fundamental a partir de este nuevo origen. Se realiza la fase de aprendizaje, que es similar a la del Linear Associator, con la diferencia que se utilizan los vectores desplazados y los valores que conforman la memoria  $M$  son tomados a partir de las componentes del vector. Aplicamos la fase de recuperación, que es similar a la que usa Lernmatrix, con la diferencia que la memoria  $M$  multiplica al vector desplazado al nuevo origen. Para finalizar con la clasificación de este vector.

## III. RESULTADOS Y CONCLUSIONES

Prueba 1.

A la primera clase pertenece el patrón

$$x^1 = \begin{pmatrix} 2.1 \\ 3.8 \end{pmatrix}$$

A la segunda clase

$$x^2 = \begin{pmatrix} 6.3 \\ 3.8 \end{pmatrix}$$

Dando como resultado la siguiente memoria, junto con los vectores desplazados respecto al centroide y clasificados correctamente.

```
Memoria

-2.1  0.0
 2.1  0.0

Vector= [ -2.1][ 0.0]
Clase:uno
Clase Resultante:uno
Vector= [ 2.1][ 0.0]
Clase:dos
Clase Resultante:dos
```

Fig. 1. Prueba 1

Prueba 2.

A la primera clase pertenece los patrones:

$$x^1 = \begin{pmatrix} 2.0 \\ 3.0 \\ 6.0 \end{pmatrix}$$

$$x^3 = \begin{pmatrix} 1.9 \\ 3.8 \\ 5.5 \end{pmatrix}$$

A la segunda clase pertenecen los patrones:

$$x^2 = \begin{pmatrix} 6.0 \\ 8.0 \\ 10.0 \end{pmatrix}$$

$$x^4 = \begin{pmatrix} 6.4 \\ 7.2 \\ 9.7 \end{pmatrix}$$

En ambas pruebas se obtuvo una recuperación perfecta, esto es debido a que el algoritmo esta preparado para clasificar de manera correcta patrones(vectores) separados entre ellos por un hiperplano. Y esto es debido al cambio de origen, por el centroide calculado, es posible que el centroide se encuentre justo en la región que separa ambas clases de patrones. Es importante hacer énfasis en que este algoritmo realiza la recuperación completa únicamente para conjuntos fundamentales con dos clases asociadas porque da por hecho que dicho conjunto esta separados linealmente.

Memoria

-6.3 -6.7 -5.9  
6.2 6.7 6.3

Vector= [ -2.1][ -2.5][ -1.8]  
Clase:uno  
Clase Resultante:uno  
Vector= [ 1.9][ 2.5][ 2.2]  
Clase:dos  
Clase Resultante:dos  
Vector= [ -2.2][ -1.7][ -2.3]  
Clase:uno  
Clase Resultante:uno  
Vector= [ 2.3][ 1.7][ 1.9]  
Clase:dos  
Clase Resultante:dos

Fig. 2. Prueba 2.

A pesar de todas las consideraciones que debe tener el conjunto fundamental, es claro ver que la aplicación de conceptos matemáticos mejora de manera importante la eficacia de algoritmos. El CAP siento un claro ejemplo donde aplicando una translación de origen impacta de manera importante en la eficacia del clasificador.