

# Escuela Colombiana de Ingeniería Julio Garavito

## Computación natural

### Selección clónica para reconocimiento de patrones (CLONALG)

Juan Andrés Moreno Silva

Escuela Colombiana de Ingeniería Julio Garavito

Estudiante de Ingeniería de Sistemas

Email: [juan.moreno-s@mail.escuelaing.edu.co](mailto:juan.moreno-s@mail.escuelaing.edu.co)

Alejandro Anzola Ávila

Escuela Colombiana de Ingeniería Julio Garavito

Estudiante de Ingeniería de Sistemas

Email: [alejandro.anzola@mail.escuelaing.edu.co](mailto:alejandro.anzola@mail.escuelaing.edu.co)

**Resumen**—En este artículo se presenta un *framework*, el cual muestra como los anticuerpos se acoplan a patrones por medio del método de selección clónica, el cual esta inspirado en el sistema inmunológico del cuerpo.

## 1. Introducción

El sistema inmune es un fuerte escudo contra amenazas externas que llegan al cuerpo, ya que cuenta con varios mecanismos de defensa los cuales están en constante evolución para garantizar la protección del mismo.

Entre los mecanismos presentes en el sistema inmunológico está el reconocimiento de entes externos, llamados antígenos<sup>1</sup>, donde, el papel del sistema es reconocer si se trata de entes benignos o entes inofensivos por medio de los anticuerpos, los cuales tienen como función de identificar los antígenos para luego desencadenar una respuesta defensiva ante la presencia de estos.

El mecanismo para que un anticuerpo reconozca un antígeno es por medio receptores, llamados *paratopos* para detectar los epítomos, los cuales son moléculas presentes en la superficie de los antígenos. Los antígenos pueden presentar varios patrones de epítomos.

**1.0.1. Afinidad.** Para este reconocimiento, una porción del antígeno (epítomo) tiene que tener una forma complementaria al receptor de la célula inmune, además no es necesario un reconocimiento perfecto para desencadenar una reacción defensiva. El grado de reconocimiento entre las dos moléculas se denomina *afinidad*.

El sistema inmunológico en su forma de reacción es en que los anticuerpos recubren al antígeno luego de determinar de que se trata de un ente que debe ser eliminado y forman un complejo alrededor, esto da señal a otras células inmunes llamadas *fagocitos*, que ingieren y digieren el complejo y destruyen al antígeno.

1. Son la porción de los patógenos que pueden promover una respuesta inmunológica adaptable.

Una de las teorías presentes en la literatura es la *teoría de selección clónica*, la cual de manera bastante aceptada por la comunidad, explica el proceso de respuesta del sistema inmune adaptable junto con la *teoría de maduración de afinidad* de los anticuerpos.

### 1.1. Teoría de selección clónica

De acuerdo a la teoría de selección clónica, el sistema inmune se compone de un conjunto de células y moléculas discretas que permanecen en reposo hasta que un agente patogénico invade al organismo. Después de la invasión, un subconjunto de estas células inmunes son capaces de reconocer el antígeno invasor y acoplarse a él. Este proceso de reconocimiento estimula las células inmunes capaces de reconocer al antígeno y empezar a reproducirse asexualmente, algunas veces llamado *clonación* o bien *expansión clónica*, dando así un subconjunto de células (clones), capaces de reconocer a ese tipo específico del antígeno.

**1.1.1. Mutación.** Como en todos los eventos reproductivos, este proceso puede estar sujeto a errores (*mutación*). Una característica particular del sistema inmune es que la mutación ocurre en proporción a la afinidad con el antígeno. Es decir, que si la afinidad entre el antígeno y el anticuerpo es alta, la tasa de mutación será baja, y si la afinidad es baja, entonces la tasa de mutación será alta.

**1.1.2. Selección.** Las células inmunes que reconocen y se acoplan satisfactoriamente con los antígenos son seleccionadas para proliferación, mientras que la células que no desempeñan ningún rol en una respuesta inmunológica mueren, y son eliminadas del repertorio de células inmunes.

En el desarrollo de este artículo se presenta un algoritmo de selección clónica que tiene como entrada una serie de imágenes con el objetivo de reconocerlas, luego de sobre llevar un entrenamiento con unas imágenes de prueba con el uso de los conceptos anteriormente presentados.

El algoritmo presentado posteriormente en este artículo se basa de los expuestos en [1] y [2].

## 2. Metodología

Se realizo un diseño inicial de un framework donde se realizo un diagrama UML<sup>2</sup> (véase la figura 1) genérico para la adaptación de lo que se desea reconocer.

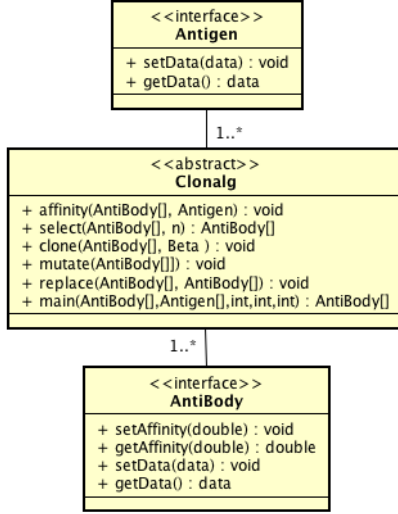


Figura 1. Diagrama UML

De donde se partió la adaptación de los elementos a reconocer explicado en la sección anterior, todo esto desarrollado en el lenguaje de programación *Java*.

### 2.1. Representación

Este *framework* cuenta con dos clases principales:

1. Anticuerpos(**Ab**): en la simulación están representados por un vector de booleanos, el cual corresponde a los píxeles de la imagen (la cual es monocromática de ahí los valores verdadero(1) y falso(0)).
2. Antígenos(**Ag**): Al igual que los anticuerpos los antígenos se representan de la misma manera.

### 2.2. Afinidad

Para esta simulación se optó por la medición de distancia Hamming, la cual consiste en contar las diferencias entre los genes del anticuerpo y el antígeno. Se define de la siguiente manera:

$$D = \sum_{i=1}^L \delta_i, \text{ donde } \delta_i = \begin{cases} 1 & \text{si } Ab_i \neq Ag_i \\ 0 & \text{si } Ab_i = Ag_i \end{cases} \quad (1)$$

Donde  $L$  es la longitud de los datos del anticuerpo. En este caso de las imágenes se tiene que  $L = w \times h$ , donde  $w$  y  $h$  son la anchura y la altura, respectivamente.

### 2.3. Población inicial

La población inicial se genera con genes aleatorios, que luego se someten a un cálculo de su afinidad respecto al antígeno y se comienza la iteración del algoritmo.

### 2.4. Selección

Antes de la selección se organizan los anticuerpos de manera ascendente de acuerdo a su afinidad; esto hace que la selección sea más sencilla ya que se seleccionan los  $n$  primeros anticuerpos.

### 2.5. Clonación

Se clonan los  $n$  mejores anticuerpos seleccionados anteriormente y se clonan inversamente proporcional al valor de su afinidad, es decir, entre menor es su afinidad, mayor número de clones tendrá, el número de clones se determina con la siguiente ecuación:

$$N_c = \sum_{i=1}^n \text{round}\left(\frac{\beta \cdot N}{i}\right) \quad (2)$$

Donde  $\beta$  es un factor multiplicador,  $N$  es el número total de anticuerpos, y  $n$  el número de anticuerpos seleccionados.

### 2.6. Mutación

En este método se toman genes al azar de los anticuerpos previamente clonados y se les invierte el valor, tenga en cuenta que la mutación es directamente proporcional a la afinidad que tenga con el antígeno seleccionado, es decir, entre mayor sea la afinidad mayor será la mutación.

### 2.7. Reemplazo

Ya que la mutación es un proceso aleatorio, esto posiblemente genere mejores anticuerpos, esto ocasiona que  $d$  anticuerpos del conjunto de anticuerpos sean reemplazados, fortaleciéndolo.

### 2.8. Algoritmo

El algoritmo empleado, basado en el algoritmo propuesto en [2] es:

Téngase en cuenta que el tamaño del conjunto  $Abm$  es igual al tamaño del conjunto  $Ag$ , es decir, cada antígeno tiene su propio anticuerpo el cual está ubicado en la misma posición.

## 3. Resultados

Se obtuvieron diferentes resultados con diferentes valores, con 30, 45 y 100 generaciones expuestas en las figuras 2, 3 y 4, respectivamente. Cabe notar que en el experimento de 100 generaciones, se finalizó en 76 generaciones, debido a que no es posible mejorar la afinidad.

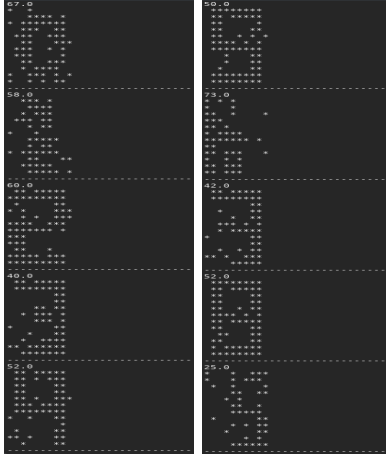


Figura 2. Aprendizaje con 30 generaciones.

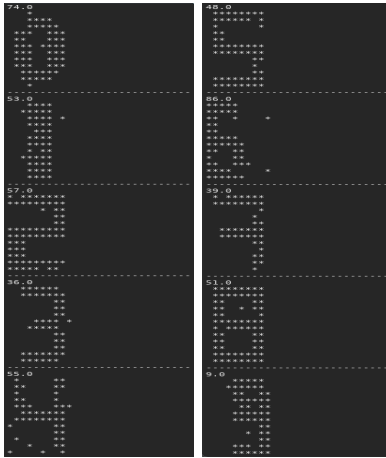


Figura 3. Aprendizaje con 45 generaciones.

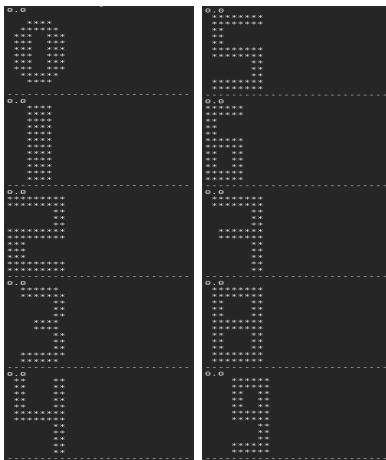


Figura 4. Aprendizaje con 100 generaciones.

---

**Algorithm 1** CLONALG( $Ab, Ag, Ngen, n, d, \beta$ )  $\rightarrow Abm$ 


---

```

1: //seleccionar aleatoriamente memoria de Ab
2: for  $g=1$  to  $Ngen$  do
3:   if  $(\forall x \in Abm : x.affinity = 0)$  then
4:     Break //ya que no pueden mejorar mas
5:   end if
6:   for  $j \in Ag$  do
7:     for  $antibody \in Ab$  do
8:        $affinity(antibody, Ag_j)$ 
9:     end for
10:     $selected \leftarrow select(Ab, n)$ 
11:     $sort(selected)$ 
12:     $cloned \leftarrow clone(selected, \beta)$ 
13:     $mutate(cloned)$ 
14:    for  $antibody \in cloned$  do
15:       $affinity(antibody, Ag_j)$ 
16:    end for
17:     $sort(selected)$ 
18:     $best \leftarrow selected_0$ 
19:    if  $Abm_j.Affinity > best.Affinity$  then
20:       $Abm_j \leftarrow best$ 
21:    end if
22:    Replace( $Ab, b$ ) //Replace los b peores anticuerpos de Ab con anticuerpos aleatorios
23:  end for
24: end for
    =0

```

---

## 4. Conclusiones

Como pudimos observar en los resultados, el algoritmo a pesar de que depende del azar, tiene buenos resultados, esto se debe a que en cada generación existe una gran posibilidad de encontrar un anticuerpo con al menos un cambio lo cual mejora la afinidad, y ya que la mutación y la selección son respecto a la afinidad, esto hace que el algoritmo converja rápidamente.

## 5. Trabajo futuro

Se planea probar este algoritmo con funciones de afinidad diferentes, para observar su comportamiento. También se tiene pensado, utilizar el algoritmo para obtener la dirección de un sonido estéreo en un espacio, en tiempo real y también utilizarlo para la detección de caracteres usando un conjunto de datos (*dataset*) y compararlo con métodos tradicionales como las *redes neuronales*.

## Bibliografía

- [1] Leandro N De Castro y Fernando J Von Zuben. “Learning and optimization using the clonal selection principle”. En: *IEEE transactions on evolutionary computation* 6.3 (2002), págs. 239-251.
- [2] Leandro Nunes De Castro. *Fundamentals of natural computing: basic concepts, algorithms, and applications*. CRC Press, 2006.