

# Aprendizaje computacional

## Aprendizaje computacional

Nelson Becerra Correa

Universidad Distrital

Fundación FABBECOR.ONG

Edgar Altamirano Carmona

Universidad Autónoma de Guerrero Fundación FABBECOR.ONG

*Fecha de recepción: 10 de diciembre de 2011*

*Fecha de aceptación: 31 de enero de 2012*

## Resumen

En el presente artículo, se hace, el diseño de un programa; que utiliza aprendizaje computacional.

Diseñamos una red neuronal que nos permite aprender un mate básico, Rey y Dama Vs Rey.

Utilizaremos el aprendizaje supervisado, Basándonos en el algoritmo de Zermelo.

Los datos generales y la estructura de la red tienen como base el trabajo del autor Nelson Becerra Correa que se presenta en el libro Técnicas de inteligencia Artificial para juegos publicado por la Universidad Distrital.

La base de datos se obtiene del trabajo de Eugenio Nalimov<sup>1</sup>.

## Abstract

In this paper, we make the design of a program, which uses computational learning.

We designed a neural network that allows us to learn a basic mate, King and Queen Vs King.

We use the supervised learning algorithm Based on Zermelo.

General data and network structure are based on the work of author Nelson Becerra Correa presented in the book Artificial Intelligence Techniques for games published by the University District.

The database is derived from the work of Eugene Nalimov.

**Key words:** Learning computer, chess, neural networks, reinforcement learning, neuron

## I. Aprendizaje

El aprendizaje computacional se refiere a las técnicas, empleadas en computación, para crear programas que aprendan a realizar una tarea de manera eficiente.

Dentro de estas técnicas podemos mencionar:

Los Árboles de decisión y regresión, Reglas de clasificación, Reglas de asociación, Programación lógica inductiva, Aprendizaje basado en grafos, Aprendizaje bayesiano y Redes bayesianas, Aprendizaje basado en instancias y casos, Clustering, Aprendizaje por refuerzo.

Existen otras técnicas de aprendizaje tales como:

Aprendizaje basado en Kernels y Support Vector Machines, Ensamblados de clasificadores, Selección de atributos, Aprendizaje se-

mi-supervisado, Algoritmos genéticos y Redes neuronales.

Otras técnicas de aprendizaje computacional han sido poco utilizadas dentro de las cuales podemos mencionar Las colonias de hormigas (*ant colony optimization* o ACO) y de enjambres (*particle swarm optimization* o PSO), Modelos lineales de regresión, Bayesianos, Modelos lineales para clasificación (funciones discriminantes, modelos generativos, modelos discriminativos, regresión logística), Modelos basados en muestreo, Procesos Gaussianos, Análisis de componentes principales, Modelos para secuencias.

Nosotros en este trabajo nos centraremos el aprendizaje supervisado, y más específicamente en las Redes Neuronales.

## II. Aprendizaje supervisado

Aprendizaje supervisado: El aprendizaje supervisado, es una técnica utilizada en ciencias de la computación, cuyo objetivo es lograr que las computadoras aprendan (Aprendizaje automático) a realizar una tarea sin la intervención de un usuario.

Dentro del aprendizaje supervisado podemos mencionar el aprendizaje por corrección de error, el aprendizaje por refuerzo y el aprendizaje estocástico.

El aprendizaje por corrección de error, ajusta los pesos a partir de los datos de salida y los datos deseados, algoritmos como perceptron, adaline y madaline y la regla delta generalizada utilizan este tipo de aprendizaje.

El aprendizaje por refuerzo, a diferencia del aprendizaje por corrección de error. No indica la salida exacta, si no que se utiliza una señal de refuerzo. Si la salida obtenida corresponde a la desea.

Para el caso del aprendizaje estocástico, se realizar cambios aleatorios y se evalúa su efecto a partir del objetivo deseado y de la distribuciones de probabilidad. La **Boltzman Machine**, ideada por Hinton, Ackley y Sejnowski en 1984 utilizan este tipo de aprendizaje igualmente **Cauchy Machine** desarrollada por Szu en 1986.

El algoritmo de aprendizaje supervisado, establece una función que relaciona los datos de entrada con los de salida.

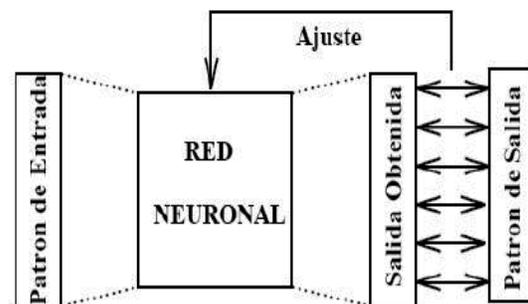
Requiere de un tutor que le indique si la salida que se obtuvo es la esperado o no. En el caso que no sea se deberá corregir el error hasta que se obtenga la salida deseada.

Fundamental mente el aprendizaje supervisado se empleado en problemas de clasificación, en el cual se establecen categorías.

Este requiere de:

- necesita un profesor que mida el funcionamiento del sistema
- maneja información de error o de control
- esta información se emplea para guiar al sistema. Hay varios algoritmos que establecen cómo se realiza esta retroalimentación, el más conocido o empleado es *elbacktracking*

Figura 1: Arquitectura de red neuronal<sup>2</sup>



## II. Estrategias en finales de juegos.

Tablas de finales, las tablas de finales se construyen a partir de lo que es conocido como "algoritmo de Inducción hacia atrás" o re-

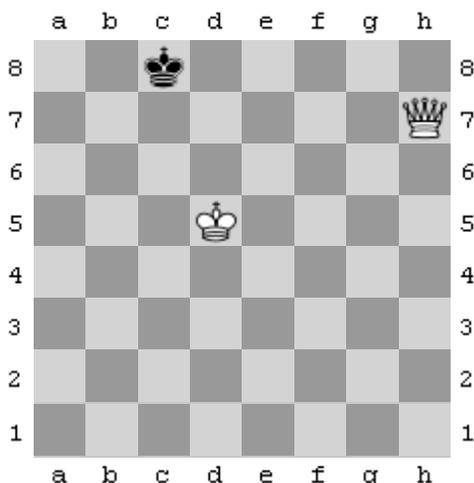
trospectivo/retrogrado, en el cual se intenta explicar que combinación de jugadas que llevaron a una determinada posición.

En el ajedrez por ejemplo: para construir las tablas de finales, se trabaja hacia atrás a partir de una posición de jaque mate. Las tablas de finales han resuelto el problema para el ajedrez con fichas que van desde 6 piezas, 5 4, 3 incluyendo los reyes.

Para algunas posiciones que se daban como tablas se probó mediante estas técnicas, que estas eran ganables, la base de datos de tablas ven los mates en 100 movimientos o más.

En otras palabras se explica la historia de la posición<sup>3</sup>

**Figura 1.** Mate Rey (♔),  
 Dama (♕) Vs Rey.



Para programar una base de datos de finales se deberá:

1. Elegir una métrica de optimalidad. La cual define en que punto de la partida el jugador ha ganado.
2. Generar todas las posiciones posibles: Procediendo de la siguiente manera:
  - a. Generar una subbase de datos que se componga de todas las posiciones en las cuales el jugador A, es ganador.
  - b. Generar una subbase de datos que se componga de todas las posiciones en las cuales el jugador B sea el ganador.
  - c. Generar una subbase de datos en la cual el jugador A sea ganador inevitable en dos movimientos.
  - d. Generar una sub base de datos en la cual el jugador B sea el ganador inevitable en dos movimientos.
  - e. Luego se generan las subbases de datos en las cuales tanto el jugador A como el jugador B, son ganadores inevitables en 3,4,5, ... , n jugadas.
  - f. Por ultimo estas subbases de datos se enlazan hacia atrás obteniendo la tablas de finales.

Por ejemplo, generar una base de datos de tablas para el final de rey y dama contra rey (RDR), hay menos de 262144 configuraciones posibles, sin embargo podemos reducir el dichas configuraciones debido a ciertas restricciones, por ejemplo, los reyes no pueden estar en casillas contiguas. Igualmente el rey enemigo no debe estar en casilla conti-

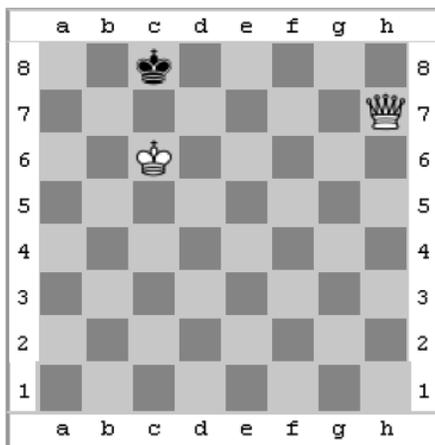
<sup>3</sup> [http://es.wikipedia.org/wiki/Base\\_de\\_datos\\_de\\_tablas\\_de\\_finales](http://es.wikipedia.org/wiki/Base_de_datos_de_tablas_de_finales) noviembre 28 de 2011

gua con la dama. De esta forma , hay al menos 40.000 posiciones legales.

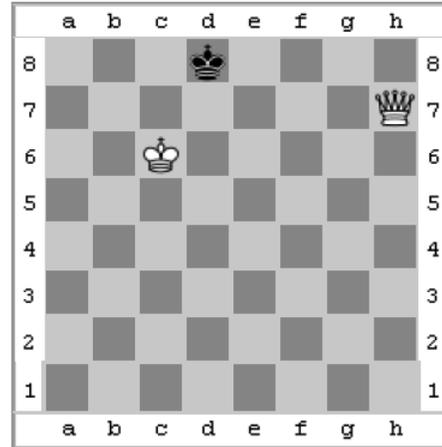
Luego de generar la base de datos y se evaluar cada posición, el resultado tiene que ser verificado independientemente. El propósito es comprobar la auto-consistencia de los resultados de la base de datos.

Por ejemplo, en la Figura 1, el programa evalúa "mate en tres (Rc6)". Entonces se observa la posición en la Figura 2, después Rc6 y se ve la evaluación "mate en dos ". Estas dos evaluaciones son consistentes la una con la otra. Si la evaluación de la Figura 2 fuera cualquier otra, sería inconsistente con la Figura 1, con lo que la base de datos necesitaría ser corregida.

**Figura 2:** Juegan Negras: mate en dos jugadas (Rd8 o Rb8)



**Figura 3:** Juegan Blancas: mate en una jugada (Dd7)



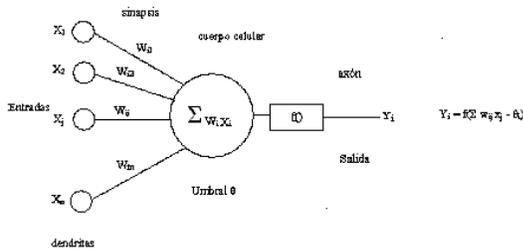
### III. Redes neuronales artificiales

Las **redes de neuronas artificiales** se pueden definir como un modelo de aprendizaje computacional inspirado en sistema nervioso de los animales. [3]

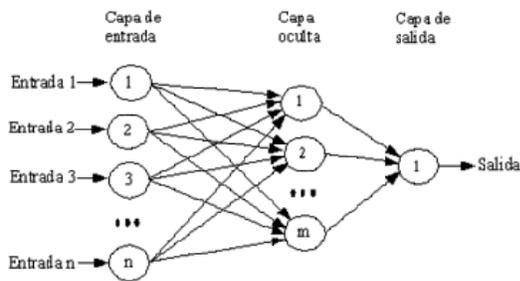
Los componentes principales de las redes neuronales artificiales

*Neurona:* conocida como unidad procesadora, esta constituida, por un conjunto de señales de entrada a las cuales se les asocia están asociados pesos (en los cuales reside el aprendizaje, de acuerdo a un ajuste). Dicha unidad procesadora transmite una señal que puede ser inhibidora o excitadora esto es debido a si se pasa un nivel llamado umbral [3].

**Figura 2: Red Neuronal de McCulloch-Pitts.**



**Figura 3: Imagen Sistema de redes neuronales multicapa**



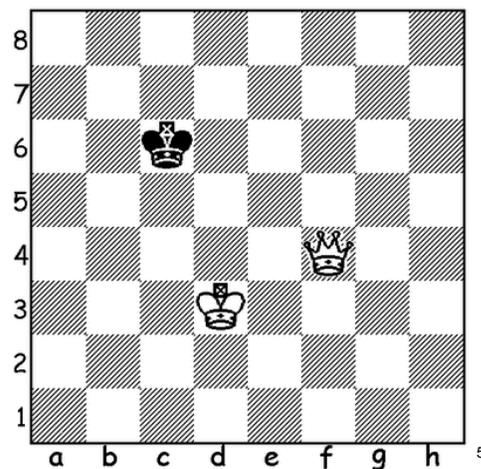
En donde  
 $w_i$  = son los pesos de la neurona  
 $X_i$  = las entradas  
 $Y_i$  = La salida  
 $+$  = corresponde a de  $x_i * w_i$   
 Umbral = indica un limite que permite a la neurona producir una señal inhibitoria, excitadora o inexistente

## IV. Diseño de una red neuronal para el aprendizaje del un mate basico en ajedrez

En este apartado se propone del diseño de una red neuronal para una final de ajedrez de tres piezas Rey, Dama contra Rey. Figura 4.

Para ello se tomo como base el trabajo de Nelson Becerra Correa, descrito en el libro: Técnicas de inteligencia artificial para juego.<sup>4</sup>

**Figura 4. Problema mate Dama, Rey Vs Rey<sup>4</sup>**



### Pasos para construir la red.

#### Formalizar el problema

- Se requiere de un arreglo de 8 posiciones para identificar las filas y otro de 8 para las columnas para la dama
- Se requiere de un arreglo de 8 posiciones para identificar las filas y otro de 8 para las columnas para el rey

4 Técnicas de inteligencia artificial para juegos, Universidad Distrital Francisco José de Caldas, Pagina...

6 Figura tomada de <http://translate.google.com/translate?hl=es&langpair=en%7Ces&u=http://chess.about.com/od/tipsforbeginners/ss/KQvsK.htm> Diciembre 2 de 2011

- Se requiere de un arreglo de 8 posiciones para identificar las filas y otro de 8 para las columnas para el rey.
  - Se requiere un arreglo de 2 posiciones para identificar
    - que ficha se juega
    - que acción se produjo
2. Mb = Secuencia de movimientos, mediante los cuales el blanco obtiene un triunfo después de n movimientos
  3. Mn = Secuencia de movimientos, mediante los cuales el negro obtiene un triunfo después de n movimientos
  4. Md = La partida termina en tablas.

**Definición de la arquitectura de red.**

Se opto por una red neuronal multicapa el perceptron por su sencillez, para el diseño e implementación de la red.

**Diseño de las entradas**

La red neuronal tendrá 24 neuronas de entrada, las cuales indican la posición para cada una de las fichas Rey, Dama y Rey.

**Diseño de salidas**

18 neuronas de salida indicando a cual posición se deberá mover y que acción se produce.

11 jaque mate, 01 mueven blancas, 10 mueven negras, 11 tablas

**Construcción de la base de datos:**

Zérmelo en 1912, demostró que todo juego bipersonal de suma cero y con información perfecta, tiene una estrategia ganadora, este se conoce como análisis retrogrado. el cual esta dado de la siguiente forma:

Algoritmo: o análisis retrogrado

1. M = Numero de movimientos validos.

Si  $M_d = M / (M_n \cup M_b)$  La partida termina en tablas.

$$M = (M_b \cup M_n \cup M_d)$$

Construir una base de datos completa para juegos como el ajedrez, go, mancala; Resulta improbable, para el ajedrez por ejemplo el número de jugadas validas es del orden de 10 a la 120 número impensable. Por tal razón es aconsejable desarrollarla única mente para finales o aperturas o problemas en particular.

Se inicia el desarrollo de la base de datos para el mate en el cual desde la posición inicial mencionada anteriormente se quiere llegar a Rey blanco en F3 fijo (no se mueve durante la partida), Dama blanca en E7 o en H6 y Rey negro en H4.

Luego desarrollamos otra parte de la base de datos para el mate Rey blanco en F3 fijo(No se mueve durante la partida), Dama blanca en E7 o en H6 y Rey negro en H4.

Por ultimo se desarrollo la parte de la base de datos la cual permite, colocar el rey negro sobre cualquier casilla de la octava fila y la dama en cualquier parte del tablero.

A continuación se presenta una estrategia ganadora parcial . Cada casilla contiene la información si la posición indica Mate, Jaque, A cual casilla jugar la dama, o si la jugada es ilegal o tablas.

**Base de datos:**

Posición inicial para el problema de ajedrez

**Tabla 1.** Estrategia ganadora para el mate, a partir de la posición  
REY BLANCO EN F3 (Rb3), DAMA BLANCA EN E7 (De7) y  
REY NEGRO EN H8 (RNKh8).

<i>Dama</i>				<b>Rey</b>					
	<b>F5</b>	<b>G5</b>	<b>G6</b>	<b>G8</b>	<b>H4</b>	<b>H5</b>	<b>H6</b>	<b>H7</b>	<b>H8</b>
<b>D6</b>		<b>E6-4</b>							
<b>E5</b>			<b>E7-7</b>	<b>F6-8</b>			<b>F6+3</b>		
<b>E6</b>					<b>H6++</b>	<b>G4-3</b>			
<b>E7</b>	<b>D6-5</b>					<b>G7-1</b>	<b>F7-5</b>		<b>G5-10</b>
<b>F5</b>									
<b>F6</b>						<b>G7-1</b>		<b>F8-7</b>	
<b>F7</b>		<b>E6-4</b>							
<b>F8</b>			<b>E7-6</b>						
<b>G4</b>							<b>G8-2</b>		
<b>G5</b>								<b>E5-9</b>	
<b>G7</b>					<b>H6++</b>				
<b>G8</b>						<b>G7-1</b>	<b>F7-5</b>		

**Tabla 2.** Base de Datos para el mate, a partir de la posición REY BLANCO EN F3 (Rbf3), DAMA BLANCA EN E7 (De7) y REY NEGRO EN CUALQUIER POCISION DEL TABLERO.

Dama	Posición Rey negro							h8
	a8	b8	c8	d8	e8	f8	g8	
a1	ilegal	tablas	a7-15	g7-16	a7-16	a7-15	f6-9	ilegal
a2	ilegal	tablas	a7-15	f7-17	a7-16	a7-15	ilegal	a7-13
a3	ilegal	tablas	a7-15	tablas	a7-16	ilegal	e7-12	a7-9
a4	ilegal	tablas	a7-15	tablas	ilegal	d7-13	f4-13	a7-13
a5	ilegal	tablas	a7-15	ilegal	c7-14	a7-15	c7-15	g5-12
a6	ilegal	a6???	ilegal	b7-15	a7-16	a7-15	f6-9	f6-13
a7	ilegal	tablas	tablas	b7-15	c7-14	d7-13	e7-12	e7-10
a8	ilegal	tablas	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal
b1	tablas	ilegal	tablas	b7-15	h7-16	b7-16	b7-14	b7-13
b2	tablas	ilegal	tablas	b7-15	h7-16	b7-16	b7-14	ilegal
b3	tablas	ilegal	tablas	b7-15	h7-16	b7-16	ilegal	b7-13
b4	tablas	ilegal	tablas	b7-15	h7-16	ilegal	b7-14	b7-13
b5	tablas	ilegal	???	b7-15	ilegal	b7-16	b7-14	b7-13
b6	tablas	ilegal	a7-15	ilegal	c7-14	b7-16	f6-9	b7-13
b7	tablas	ilegal	ilegal	f7-17	c7-14	d7-13	e7-12	e7-10
b8	tablas	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal
c1	tablas	tablas	ilegal	c6-19	c7-14	c7-17	c7-14	g5-9
c2	tablas	tablas	ilegal	h7-16	c7-14	h7-15	c7-14	c7-13
c3	tablas	tablas	ilegal	g7-16	c7-14	c7-17	f6-9	ilegal
c4	tablas	tablas	ilegal	f7-16	c7-14	c7-17	ilegal	c7-13
c5	tablas	tablas	ilegal	c6-19	c7-14	ilegal	e7-12	e7-11
c6	ilegal	tablas	ilegal	b7-15	ilegal	d7-13	f6-9	c7-13
c7	tablas	ilegal	ilegal	ilegal	a7-16	d7-13	e7-12	e7-12
c8	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal
d1	tablas	tablas	tablas	ilegal	d6-19	d7-13	d7-16	d7-13
d2	tablas	tablas	tablas	ilegal	d6-19	d7-13	d7-16	d7-13
d3	tablas	tablas	tablas	ilegal	h7-16	d7-13	d7-16	d7-13
d4	tablas	tablas	tablas	ilegal	a7-16	d7-13	f6-9	ilegal
d5	ilegal	tablas	tablas	ilegal	d6-19	d7-13	ilegal	d7-13
d6	tablas	ilegal	tablas	ilegal	c7-14	ilegal	f6-9	e7-10
d7	tablas	tablas	ilegal	ilegal	ilegal	a7-15	e7-12	e7-9
d8	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal
e1	tablas	tablas	tablas	c3?????	ilegal	e6-12	e7-12	e7-12
e2	tablas	tablas	tablas	c4?????	ilegal	e6-12	e7-12	e7-12
e3	tablas	tablas	tablas	c5?????	ilegal	e6-12	e7-12	e7-12
e4	ilegal	tablas	tablas	b7-15	ilegal	e6-12	e7-12	e7-12
e5	tablas	ilegal	tablas	g7-17	ilegal	e6-12	f6-10	ilegal
e6	tablas	tablas	ilegal	f7-17	ilegal	d7-13	ilegal	e7-12
e7	tablas	tablas	tablas	ilegal	ilegal	ilegal	f6-10	1raWq. g5-10
e8	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal
f1	tablas	a6???	b5??	c4?????	d3?????	e2??	e2??	g2-12
f2	tablas	tablas	a7-16	c5?????	a7-16	a7-15	a7-14	g2-12
f3	ilegal	ilegal	ilegal	ilegal	c7-14	ilegal	ilegal	ilegal
f4	tablas	ilegal	b4???	f7-17	h7-15	ilegal	f6-10	g3-12
f5	tablas	tablas	ilegal	f7-17	g7-16	ilegal	f6-10	g4-12
f6	tablas	a6???	b6??	ilegal	ilegal	ilegal	e7-12	ilegal
f7	tablas	tablas	a7-16	b7-15	ilegal	ilegal	ilegal	e7-10
f8	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal
g1	tablas	tablas	tablas	c5?????	g7-16	a7-15	ilegal	g4-12
g2	tablas	tablas	b2??	g7-17	g7-16	e2??	ilegal	g4-12
g3	tablas	ilegal	tablas	g7-17	g7-16	c7-17	ilegal	g4-12
g4	tablas	tablas	ilegal	g7-17	g7-16	d7-13	ilegal	g3-12
g5	tablas	tablas	b5??	ilegal	g7-16	e5??	ilegal	g3-12
g6	tablas	tablas	b6??	g7-17	ilegal	e6??	ilegal	g3-12
g7	tablas	tablas	a7-16	b7-15	c7-14	ilegal	ilegal	ilegal
g8	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal
h1	tablas	tablas	b1??	h7-17	h7-16	h7-15	e1??	ilegal
h2	tablas	ilegal	b2??	h7-17	h7-16	h7-15	c7-14	ilegal
h3	tablas	tablas	ilegal	h7-17	c7-14	d7-13	d7-16	ilegal
h4	tablas	tablas	b4??	ilegal	h7-16	h7-15	f6-10	ilegal
h5	tablas	Tablas	b5??	h7-17	ilegal	h7-15	f4-15	ilegal
h6	tablas	Tablas	b6??	h7-17	h7-16	ilegal	f6-10	ilegal
h7	tablas	Tablas	a7-16	a7-16	c7-14	d7-13	ilegal	ilegal
h8	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal	ilegal

## Estructura de la red neuronal

Para poder identificar la posición de la ficha dentro del tablero se estructura un arreglo de ocho posiciones para identificar, la fila en la cual esta la ficha y otro para las columnas, el contenido de cada arreglo se inicializa en cero y la posición que corresponda a la ficha se le da el valor de uno, por consiguiente el arreglo tendrá valores boléanos. Adicionalmente, se requiere otro arreglo de dos posiciones para indicar, el resultado de la jugada, tablas, ilegal, mate o jaque , codificadas asi:

- 0 0 0 Juegan negras
- 0 0 1 Juegan blancas
- 0 1 0 Jaque
- 1 0 0 Jugada ilegal
- 1 1 1 Jaque Mate

Se toma como referencia un tablero estructurado de la siguiente manera.

Arreglo para las filas

0	0	0	0	0	0	0	0
A	B	C	D	E	F	G	H

Arreglo para las columnas

0	0	0	0	0	0	0	0
A	B	C	D	E	F	G	H

Para indicar que una ficha esta en la casilla F3.

Tendríamos,

0	0	1	0	0	0	0	0
0	0	0	0	0	1	0	0

Se requiere dos arreglos uno que indica las filas y otro las columnas por cada ficha presente en la final del juego de ajedrez.

Por consiguiente, la codificación para el mate

Seria :

**Figura 6.** Estrategia ganadora para el mate rey , dama vs rey

DAMA BLANCA		REY BLANCO		REY NEGRO		JUGADA		ACCION
FILA	COL	FILA	COL	FILA	COL	FILA		

```

00010000 00000100 00000010 00001000 00001000 00000100 00
00001000 00001000 00000010 00000001 00000100 00000100 00
00001000 00001000 00000010 00000100 00001000 00000010 00
00001000 00001000 00000001 00000100 00000100 00000100 00
00001000 00000100 00000001 00001000 00000010 00010000 10
00001000 00000100 00000001 00010000 00000001 00000100 11
00001000 00000010 00000010 00001000 00010000 00000100 00
00001000 00000010 00000001 00000001 00000010 00001000 00
00001000 00000010 00000001 00001000 00000010 00000010 00
00001000 00000010 00000001 00000100 00000100 00000010 00
00000100 00000100 00000001 00010000 00000001 00000100 11
00000100 00000100 00000001 00001000 00000010 00000010 00
00000100 00000100 00000001 00000010 00000100 00000001 00
00000100 00000010 00000001 00000010 00001000 00000100 00
00000100 00000001 00000010 00000010 00001000 00000010 00
00000010 00010000 00000001 00000100 00000010 00000001 00
00000010 00001000 00000001 00000010 00001000 00001000 00
00000010 00000001 00000010 00000010 00001000 00000010 00
00000010 00000001 00000001 00000001 00010000 00000001 00000100 11
00000010 00000001 00000001 00001000 00000010 00000010 00
00000010 00000001 00000001 00000001 00000100 00000010 00
    
```

1. Selección del modelo de red neuronal.

Para implementar el problema anterior se escogió una red con las siguientes características.

- Topología, multicapa con conexión hacia delante (Feedforward)
- Mecanismo de aprendizaje supervisado.

Un modelo de red que reúne estas características es el PERCEPTRON MULTICAPA, para tal efecto se escogió una red desarrollada por Valluru Rao y Hayagriva Rao, en el libro C++ Neuronal Networks and Fuzzy Logic, Segunda Edición.

A este programa se le hizo una migración a C++ Builder y se adaptó el modo gráfico para que permitirá a un adversario humano jugar contra el computador.

### 4.3 Resultados Experimentales

La tabla 1. muestra el resultado del entrenamiento de la red neuronal para el mate mencionada, en el problema iniciando en la posición:

Como se aprecia en la tabla 1. la mejor configuración se presenta cuando se tiene 32 neu-

ronas de entrada, 24 neuronas ocultas y 18 de salida con un parámetro de aprendizaje de 0.5. sin embargo basta tener una configuración de 32 neuronas de entrada, 6 ocultas y 18 de salida para que la red adquiriera un correcto aprendizaje. La Zona demarcada con Azul muestra que la red está lo suficientemente entrenada para solucionar el problema y la zona roja es la configuración ideal.

primera capa.

-0.720000 Peso de conexión entre la primera neurona de la primera capa y la primera neurona de la segunda capa, -0.680000 Peso de conexión entre la primera neurona de la primera capa y la segunda neurona de la segunda capa,..., 0.840000 Peso de conexión entre la primera neurona de la primera capa y la última neurona de la segunda capa.

Luego de entrenar la Red Neuronal, y ponerla en ejecución mediante un archivo de jugadas se obtuvieron los siguientes resultados.

En la tabla 3 se aprecia el vector de entrada, para la red neuronal y el vector de salida o sea la respuesta de la red neuronal.

**Tabla 3.** Resultados finales de la red neuronal

Para el vector de entrada:									
0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	0.000000	0.000000	0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000
0.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000
El vector de salida es									
0.000032	0.000009	0.000007	0.000365	0.999360	0.000027	0.000013	0.000090	0.000011	0.000007
0.000007	0.000017	0.000001	0.000000	0.999791	0.000184	0.000001	0.000067	0.000041	0.000000

Son 32 neuronas de entrada, distribuidas así:

```
0.000000 0.000000 0.000000 1.000000
0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000
0.000000 1.000000 0.000000 0.000000
0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 1.000000 0.000000
0.000000 0.000000 0.000000 0.000000
1.000000 0.000000 0.000000 0.000000
```

La primer fila corresponde a la ubicación de la Dama Blanca en la Columna del tablero.

```
0.000000 0.000000 0.000000 1.000000
0.000000 0.000000 0.000000 0.000000
```

La segunda fila corresponde a la ubicación de la Dama Blanca Fila del tablero.

```
0.000000 0.000000 0.000000 0.000000
0.000000 1.000000 0.000000 0.000000
```

En donde, la dama blanca se ubica la **columna D** y **fila 6 (D6)**.

La primer fila corresponde a la ubicación de el Rey Negro en la Columna del tablero.

```
0.000000 0.000000 0.000000 0.000000
0.000000 0.000000 1.000000 0.000000
```

La segunda fila corresponde a la ubicación del Rey Negro en la Fila del tablero.

```
0.000000 0.000000 0.000000 0.000000
1.000000 0.000000 0.000000 0.000000
```

En donde, el Rey Negro se ubica la **columna G** y **fila 5 (G5)**.

Luego la Red Neuronal responde con :

```
0.000032 0.000009 0.000007 0.000365
0.999360 0.000027 0.000013 0.000090
0.000011 0.000007 0.000017 0.000001
0.000000 0.999791 0.000184 0.000001
0.000067 0.000041
```

Lo cual indica que la Dama Blanca debe colocarse en la Columna E, 0.999360, pues es el valor mas cercano a uno.

La Fila correspondiente será 4, 0.999791 valor próximo a uno. Por lo cual debe mover a 4.

Las dos ultimas neuronas indican por su valor próximo a cero, que deben jugar Negras 0.000067 0.000041

Comparando con la Base de Datos<sup>7</sup> tenemos que para la posición Dama Blanca En D6 y Rey Negro en G5. La respuesta debe ser E6.

Un análisis similar se debe hacerse para los demás elementos de la tabla.

*Modificaciones respecto a la red anterior*

## 4.5 Conclusiones

Para implementar un juego mediante red neuronal, deberá primero utilizarse una red sin aprendizaje supervisado pues es imposible diseñar una base de datos para un juego en total.

Se requiere un arreglo de 16 neuronas por cada ficha presente dentro del tablero de juego, ocho para indicar la fila en la cual se encuentra la ficha y ocho para las columnas, además se requiere de una neurona adicio-

<sup>7</sup> Base de Datos diseñada para el mate y mostrada en la Tabla 1.

nal para indicar el color de la ficha y dos para la acción a tomar.

Por consiguiente para las 32 figuras del tablero se requerirían un total de 256 neuronas de entrada mas 1 para indicar el color y 2 para indicar la acción a tomar. Para un total de 259 Neuronas, de entrada.

Para entrenar una red neuronal para un juego como el ajedrez, es bastante improbable por consiguiente su utilización debe dedicarse a aperturas y finales de juego, lo cual ahorra tiempo pues se evita generar un árbol de juegos.

### Trabajos futuros

Ensayar la apertura Ruy López. Generando una base de datos.

Como trabajo adicional seria la creación de un programa de juegos bipersonales de suma cero , información perfecta. Utilizando cualquiera de las siguientes técnicas.

Programación molecular, Programación con ADN, Bio-computacion, Computación cuántica.

**Tabla 3.** Resultado del entrenamiento de la Red Neuronal, para el aprendizaje del mate.

Neuronas de entrada	Neuronas ocultas	Neuronas Salida	Parámetro aprendizaje	Promedio error por ciclo	Error después del ultimo ciclo	Error por parámetro después del ultimo ciclo	Total ciclos	Total de Parámetros
32	1	18	0.5	3.640406	4.492844	1.00463	1000	24000
32	2	18	0.5	3.033418	7.493486	1.675594	1000	24000
32	4	18	0.5	1.880221	1.328615	0.297087	1000	24000
32	6	18	0.5	0.494747	0.021927	0.004903	1000	24000
32	8	18	0.5	0.410736	0.009835	0.002199	1000	24000
32	10	18	0.5	0.421034	0.007328	0.0011639	1000	24000
32	24	18	0.5	0.496901	0.004465	0.000998	451	830
32	1	18	0.25	2.909894	3.127119	0.699245	1000	24000
32	2	18	0.25	2.839579	2.830108	0.632831	1000	24000
32	4	18	0.25	1.258927	0.61162	0.136762	1000	24000
32	6	18	0.25	0.69099	0.038017	0.008501	1000	24000
32	8	18	0.25	0.490276	0.023047	0.005153	1000	24000
32	10	18	0.25	0.499981	0.013703	0.003064	1000	24000
32	24	18	0.25	0.336127	0.004494	0.001005	1000	24000

### Bibliografía:

- [1] Nelson Becerra Correa. Técnicas de inteligencia artificial en Juegos, Universidad Distrital Francisco José de Caldas (2003).
- [2] D. Patterson. Artificial Neural Networks. Theory and Applications. Prentice Hall Inc. (1995).