

TALLER II INTELIGENCIA ARTIFICIAL

1. Resumen

La Inteligencia Artificial (IA) es probablemente una de las ramas de las Ciencias de la Computación que más crecimiento está teniendo en la actualidad. Pese a haber nacido hace más de 70 años, se encuentra en el periodo de su historia en el que mayor interés ha generado debido a la revolución que está provocando en el mercado actual.

Hasta hace poco, existía una limitación de capacidad de cómputo que hacía que la Inteligencia Artificial proporcionara resultados muy pobres en los problemas en los que se aplicaba, lo que produjo varios periodos históricos de descontento en la industria y una considerable reducción tanto del interés en esta disciplina como del número de investigadores dedicados.

La inteligencia artificial (IA) es una expresión técnica referida a artefactos empleados para detectar contextos o llevar a cabo acciones en respuesta a contextos detectados. Nuestra capacidad de construir dichos artefactos ha aumentado y, con ello, el impacto que tienen en nuestra sociedad

Esta capacidad de descubrimiento de la IA se ha ampliado no solo por el incremento masivo de los datos digitales y la capacidad de computación, también por las innovaciones en los algoritmos de IA y AA. Hoy buscamos fotografías, vídeos y audio (Barrett et al., 2016; Wu et al., 2016). Podemos traducir, transcribir, leer labios, interpretar emociones (incluidas las mentiras), falsificar firmas y otros tipos de escritura manual y manipular vídeos

2. Palabras Claves

Perceptrón, Red neuronal, Aprendizaje, Automatización, Algoritmo, Python,

Lógica, Resultados, Errores, Pendiente, Calculo, Analizar, Optimizar.

3. Summary

Artificial Intelligence (AI) is probably one of the branches of Computer Science that is having more growth today. Despite being born more than 70 years ago, it is in the period of its history in which it has generated more interest due to the revolution it is causing in the current market.

Until recently, there was a limitation of computing capacity that made Artificial Intelligence provide very poor results in the problems in which it was applied, which produced several historical periods of discontent in the industry and a considerable reduction in both interest in this discipline as of the number of dedicated researchers.

Artificial intelligence (AI) is a technical expression referring to artifacts used to detect contexts or carry out actions in response to detected contexts. Our ability to build such artifacts has increased and, with it, the impact they have on our society

This ability to discover AI has been extended not only by the massive increase in digital data and computing capacity, but also by innovations in AI and AA algorithms. Today we look for photographs, videos and audio (Barrett et al., 2016; Wu et al., 2016). We can translate, transcribe, read lips, interpret emotions (including lies), falsify signatures and other types of handwriting and manipulate videos

4. Keywords

Perceptron, Neural network, Learning, Automation, Algorithm, Python, Logic, Results, Errors, Slope, Calculation, Analyze, Optimize.

5. Introducción

La Inteligencia Artificial (IA), es la rama de la ciencia, la cual se centra en el desarrollo y creación de programas y/o máquinas que imitan y exponen una inteligencia y comportamiento humano. Esta IA, está compuesta por diversas

ramas o campos, entre ellas tenemos las redes neuronales artificiales, las cuales son modelos matemáticos, inspirados en el comportamiento de las neuronas biológicas y en la forma que se organizan formando la estructura de cerebro, conectadas entre sí, trabajando en conjunto, sin que se tenga una tarea en concreta para cada una, pero que va creando y reforzando las conexiones, para ir aprendiendo cada vez más. Algunos ejemplos, como son la simple, la múltiple o el perceptrón, de los cuales se analizaran, realizando diversos ejercicios, evaluando sus resultados y sacando conclusiones en cuanto a estos.

6. Algoritmos Usados

Perceptrón, Red Neuronal Simple, Red Neuronal Múltiple, Backpropagation.

7. Lenguaje de Programación Usado

Para la realización del taller usamos el lenguaje de programación Python, el cual es un lenguaje interpretado, capaz de analizar y ejecutar otros programas, el cual posee una sintaxis que favorece el código legible. Este lenguaje de programación es multiparadigma, ya que soporta diferentes tipos de programación, adicional es multiplataforma y un lenguaje dinámicamente tipado ya que una misma variable puede tomar valores de distinto tipo en distintos momentos.

De igual manera usamos el paquete TKINTER, el cual es un paquete multiplataforma estándar de Python para crear interfaces gráficas de usuario (GUI).

Proporciona acceso a un intérprete de Tcl subyacente con el kit de herramientas Tk, que en sí mismo es una biblioteca de interfaz de usuario gráfica multiplataforma y multiplataforma.

8. Códigos Algoritmos Implementados

PERCEPTRON

```
from random import choice

from numpy import array, dot, random

import matplotlib.pyplot as plt


# Funcion de Activación

activacion = lambda x:0 if x < 0 else 1


# Set de Entrenamiento

entrenamiento = [

    (array([0,0,1]),0),

    (array([0,1,1]),1),

    (array([1,0,1]),1),
```

```
(array([1,1,1]),1),  
]
```

```
w = random.rand(3)
```

```
errores = []
```

```
esperados = []
```

```
bahias = 0.2
```

```
n = 100
```

```
# Entrenamiento
```

```
for i in range(n):
```

```
    x, esperado = choice(entrenamiento)
```

```
    resultado = dot(w, x)
```

```
    esperados.append(esperado)
```

```
    error = esperado - activacion(resultado)
```

```
    errores.append(error)
```

```
# Ajuste
```

```
w += bahias * error * x
```

```
for x, _ in entrenamiento:
```

```
    resultado = dot(w, x)
```

```
    print("{ } : { } -> { } ".format(x[:3], resultado, activacion(resultado)))
```

```
[0 0 1]: -0.07397738495418188 -> 0
```

```
[0 1 1]: 0.14702093515346765 -> 1
```

```
[1 0 1]: 0.8998619915145738 -> 1
```

```
[1 1 1]: 1.1208603116222233 -> 1
```

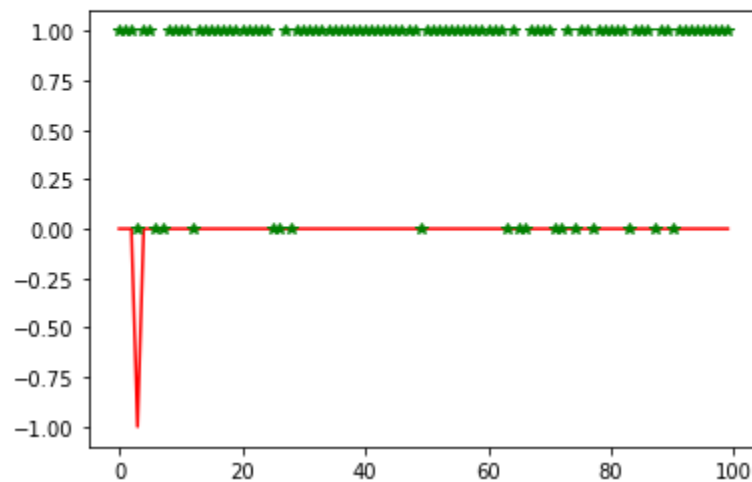
```
plt.plot(erroses, '-', color='red')
```

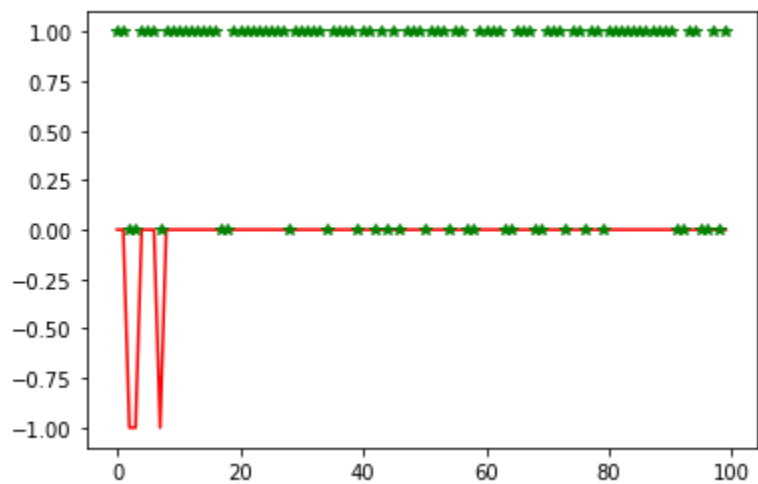
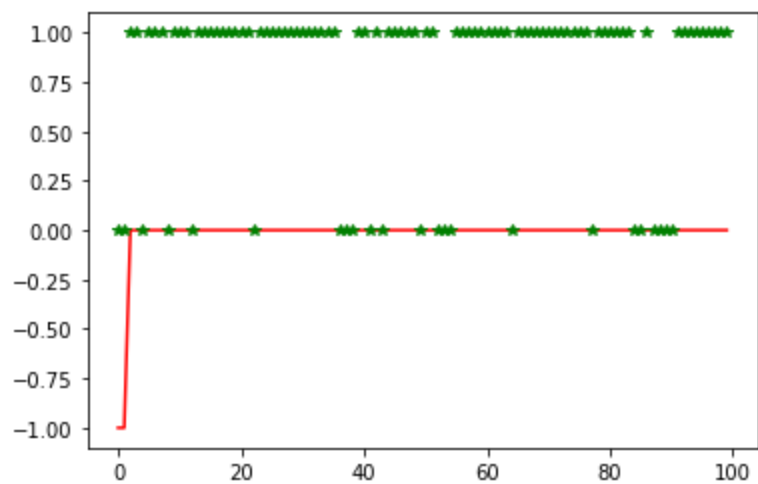
```
plt.plot(esperados, '*', color='green')
```

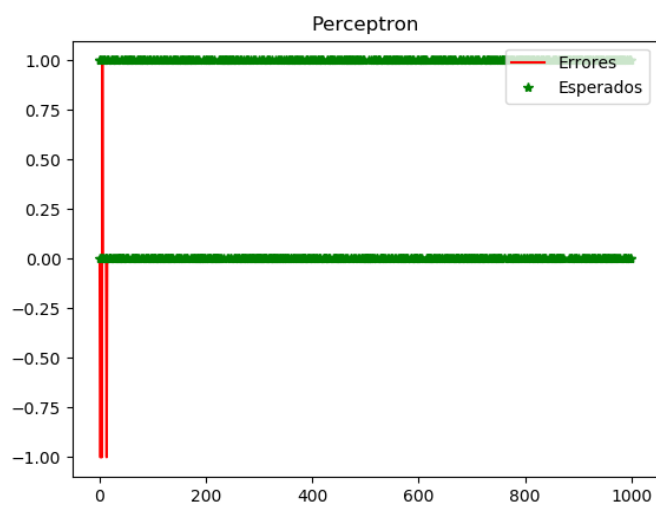
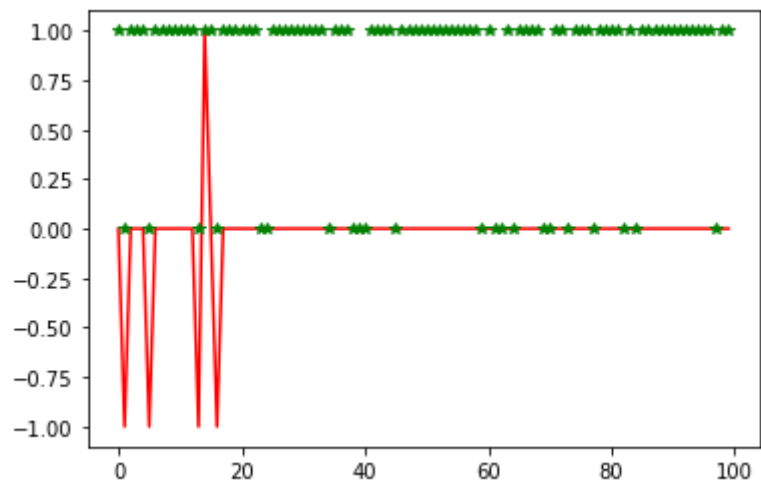
RESULTADOS ESPERADOS

```
In [16]: plt.plot(erroses, '-', color='red')
plt.plot(esperados, '*', color='green')
```

```
Out[16]: [<matplotlib.lines.Line2D at 0x9087c10>]
```







Perceptrón	Red neuronal simple	Red neuronal multicapa
------------	---------------------	------------------------

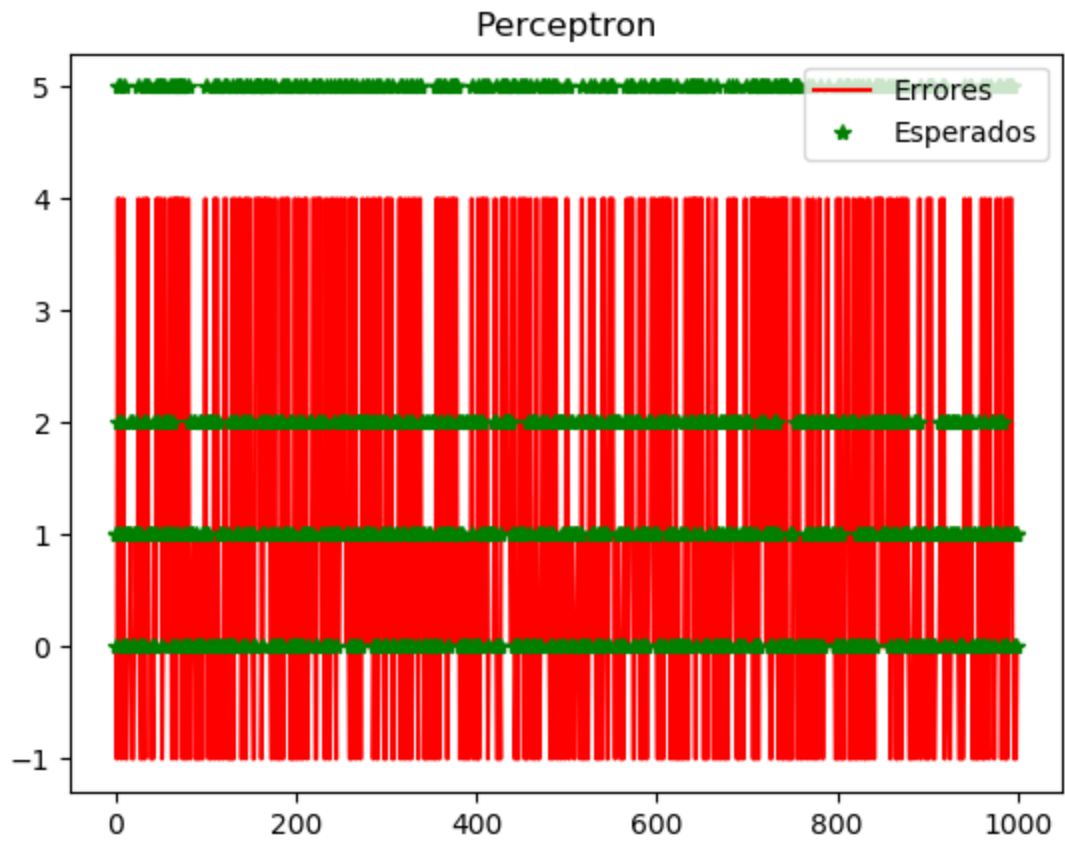
Ingrese la misma cantidad de ítem para cada entrada, separados por coma y sin espacio.

Entrada # 0 :	<input type="text" value="0,0,1"/>	Salida # 0 :	<input type="text" value="5"/>
Entrada # 1 :	<input type="text" value="1,1,1"/>	Salida # 1 :	<input type="text" value="2"/>
Entrada # 2 :	<input type="text" value="1,0,1"/>	Salida # 2 :	<input type="text" value="1"/>
Entrada # 3 :	<input type="text" value="0,1,1"/>	Salida # 3 :	<input type="text" value="0"/>

Ingrese el sesgo de bahias	<input type="text" value="0.2"/>
Ingrese entrenamiento	<input type="text" value="1000"/>

Entrenar

[0.0.1.]: 199.9871368818879 -> 1 0
 [1.1.1.]: 252.31194852322295 -> 1 .
 [1.0.1.]: 250.20582316592578 -> 1 .
 [0.1.1.]: 202.0932622391851 -> 1 .



Perceptrón	Red neuronal simple	Red neuronal multicapa
Ingrese la misma cantidad de item para cada entrada, separados por coma y sin espacio.		
Entrada # 0 :	<input type="text" value="1,1,1"/>	Salida # 0 : <input type="text" value="0"/>
Entrada # 1 :	<input type="text" value="0,1,1"/>	Salida # 1 : <input type="text" value="1"/>
Entrada # 2 :	<input type="text" value="1,1,0"/>	Salida # 2 : <input type="text" value="1"/>
Entrada # 3 :	<input type="text" value="0,0,1"/>	Salida # 3 : <input type="text"/>
Ingrese el sesgo de bahias	<input type="text" value="0.7"/>	
Ingrese entrenamiento	<input type="text" value="1000"/>	
<input type="button" value="Entrenar"/>		

RED NEURONAL SIMPLE

```

from numpy import exp, array, random, dot

class RedNeuronal():
    def __init__(self):
        self.pesos_signaticos = 2 * random.random((3,1)) - 1

    def __sigmoide(self, x):
        return 1 / (1 + exp(-x))

    def __sigmoide_derivado(self, x):
        return x * (1 - x)

    def entrenamiento(self,entradas,salidas,numero_iteraciones):
        for i in range(numero_iteraciones):
            salida = self.pensar(entradas)
            error = salidas - salida
            ajuste = dot(entradas.T, error * self.__sigmoide_derivado(salida))
            self.pesos_signaticos += ajuste

    def pensar(self,entrada):
        return self.__sigmoide(dot(entrada, self.pesos_signaticos))

if __name__ == '__main__':
    red_neuronal = RedNeuronal()
    entradas = array([[0,0,1], [1,1,1], [1,0,1], [0,1,1]])
    salidas = array([[0,1,1,0]]).T
    red_neuronal.entrenamiento(entradas,salidas,1000)
    print(red_neuronal.pesos_signaticos)
    print(red_neuronal.pensar(array([1,0,0])))

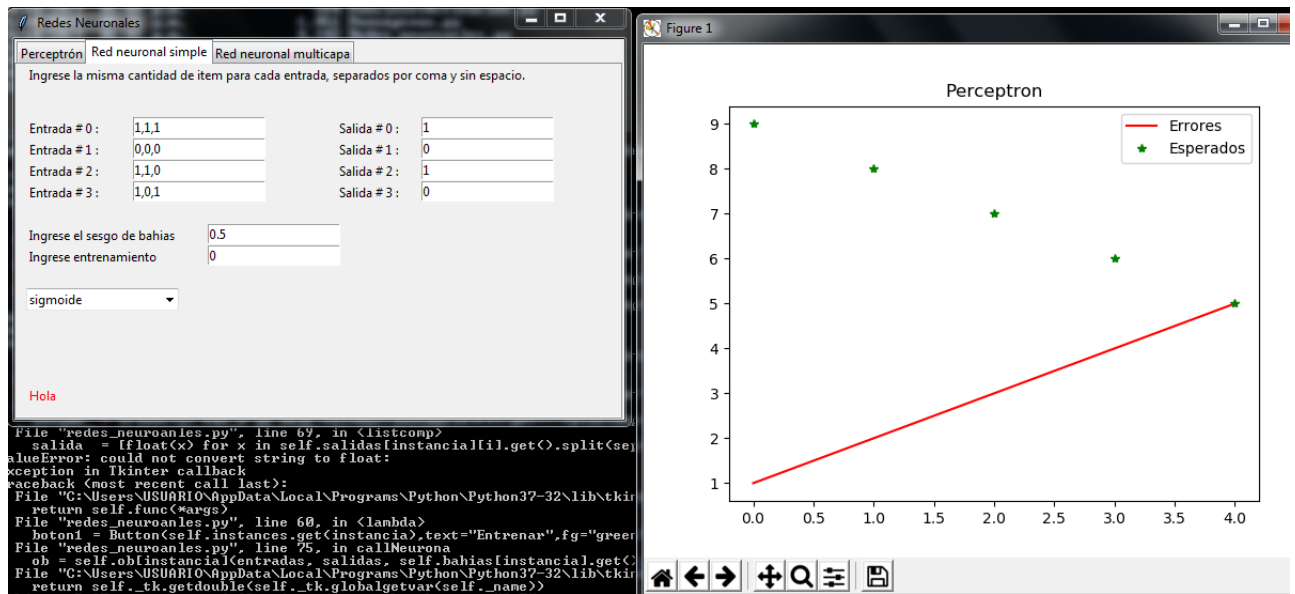
```

```
[[ 7.25755115]
```

```
[-0.22243361]
```

```
[-3.41062564]]
```

```
[0.99929566]
```



9. Resultados Alcanzados

Gracias a las herramientas visuales, y teoría que ofrece el internet y con la adecuación de cada uno de estos algoritmos de redes neuronales, logramos comprender un poco más cada uno de los conceptos de la IA, además conocer más el funcionamiento y el comportamiento de estas complejas redes.

10. Conclusiones

La inteligencia artificial es una ciencia que recién surge en estas últimas décadas y la cual está cambiando la forma en que los humanos hacemos las cosas, incluso las más cotidianas. Debemos estar mentalizados que estas innovaciones por más que se parezcan al hombre, siempre dependerán del mismo para poder evolucionar, de esta manera el hombre tampoco debe sentirse desplazado, por lo contrario puede llegar a ser la forma de podernos liberar.

Las redes neuronales nos brindan la suficiente información para determinar cuan grande podría ser el alcance de esta ciencia aplicada a los diferentes ámbitos de la sociedad, como lo son la medicina, el campo, la informática, la seguridad, mercados, militar, etc. Las redes neuronales tienen grandes ventajas al ser implementadas para solventar una necesidad en la Ingeniería, pues toman

como referente principal el funcionamiento biológico del cerebro y mediante estudios puede guiarse para su análisis, teniendo como gran apoyo la gran diversidad de modelos matemáticos.

12. Recomendaciones

Estas herramientas son muy útiles para aplicar en la automatización de máquinas, domótica y robótica, y facilitar sus acciones, pasando por un proceso de entrenamiento para que al pasar del tiempo las hagan con mayor facilidad y naturalidad, pero se debe tener cuidado al momento de implementar estos algoritmos, ya que un mal parámetro, una instrucción inadecuada, puede dar malos resultados.

Para profundizar en este mundo del machine learning se deben tener muchos complementos matemáticos y fundamentos de programación, para lograr un buen entendimiento y explotar esto al máximo.

13. Computadores Usados

- **Equipo 1:** Intel Core 2 Duo 2.26 GHz y 3 GB RAM.

14. Referencia Bibliográficas

-
-