

**Inteligencia Artificial**

**Trabajo 2**

**Integrantes**

Renzo Eduardo Damian Gomez U201715758

Luis Enrique Guadalupe Carrillo U201716383

**Sección**

SI62

**Profesor**

Wester Edison Zela Moraya

**Ciclo Académico**

2020-01

Índice

I. Descripción y fundamentación del problema

II. Descripción del data set para la red neuronal

III. Descripción de la arquitectura de Red Neuronal y su salida

IV. Detalles del código

V. Pruebas de uso

VI. Referencias

1. **Descripción y fundamentación del problema**

Gunbound es un videojuego multijugador en el cual 2 o más adversarios conduciendo vehículos de guerra se enfrentan en distintos escenarios en los cuales tienen que derrotar al resto destruyendolos lanzando misiles hasta dañar por completo el vehículo enemigo. Este popular juego del 2003 se caracterizó porque a pesar de ser los mismos escenarios y los mismos protagonistas, las partidas nunca eran iguales, siempre variaba pues, a pesar de ser un jugador experto y dominar todos los mapas, era casi imposible siempre saber la mira correcta (combinación entre la potencia y el ángulo del disparo).

Entonces, la principal motivación viene a ser la aplicación de inteligencia artificial en el aprendizaje de los bots en los videojuegos, por lo que nuestra propuesta es desarrollar un software (videojuego), que con la implementación de IA, mediante uso de una red neuronal, permita al bot de un videojuego aprender en cada movimiento que este realice. Basándonos en este popular juego nuestro trabajo consta de crear un bot (carrito) que optimice su lanzamiento, con un movimiento parabólico, lanzando proyectiles cada vez más cerca a la ubicación del enemigo.

Finalmente, en este “Trabajo 2” se resolverá un problema en particular, que, en un escenario con una sola plataforma plana, la computadora (bot) será capaz de lanzar el proyectil “perfecto” con el uso de una red neuronal y un data set que brindará de información al modelo computacional para realizar el cometido. Además de la aplicación del perceptrón multicapa con 1 capa de neuronas ocultas que, en este caso, es una de las redes más potentes y eficientes en esta área de aplicación. “Dentro del marco de las redes de neuronas, el Perceptron multicapa es en la actualidad una de las arquitecturas más utilizadas en la resolución de problemas” (2017). Cuya arquitectura se explicará mejor en el siguiente punto.

1. **Descripción del data set para la red neuronal**

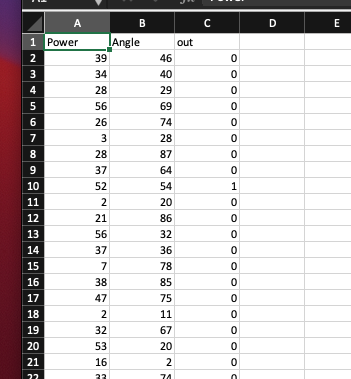


Fig.1

La Fig.1 muestra los datos recopilados de Poder, Angulo y Resultado en un total de 10 000 casos, donde “0” indica que no llegara el proyectil y “1” que sí.

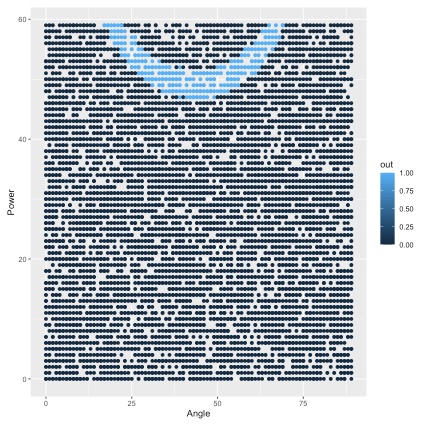


Fig.2

Como se ve en la Fig.2, se tiene el gráfico de 10 000 datos recopilados de Angulo vs Poder (Angulo:0-90° y Poder: 0-60) en el que se analiza los outputs obtenidos del dataset recopilado.

Cada punto es el output que se obtiene, mientras más cercano esté el proyectil lanzado del objetivo se considera que se obtiene un mejor resultado (Negro: muy lejos del objetivo, Celeste: Llegó al objetivo).

Finalmente, los proyectiles que lograron atinar al objetivo exitosamente son los que eventualmente formaron una “U” celeste.

1. **Descripción de la arquitectura de Red Neuronal y su salida**

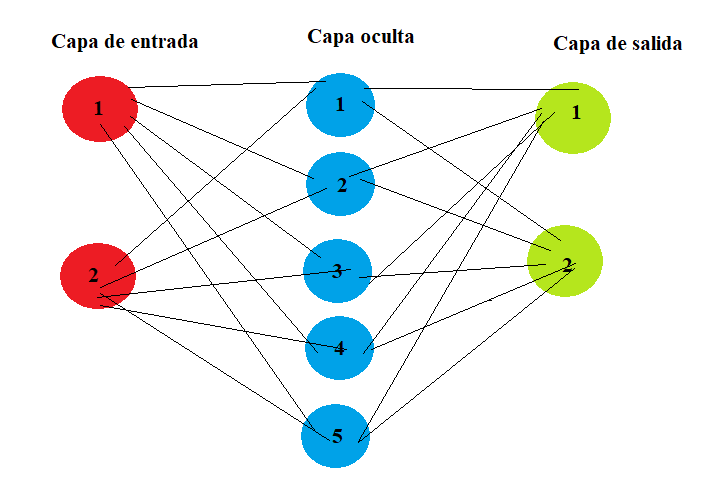
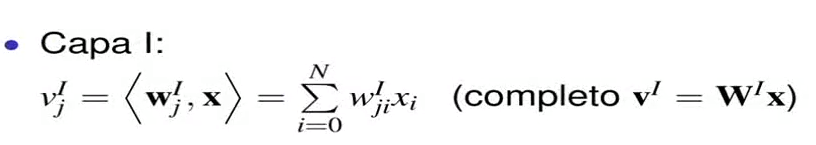
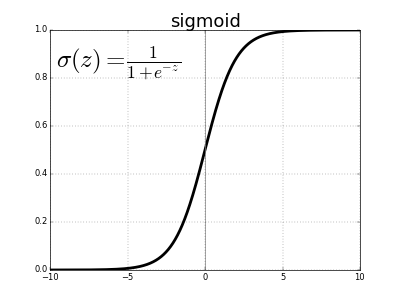


Fig.3

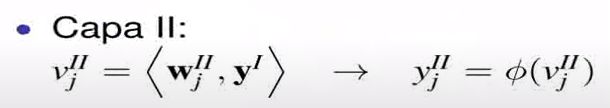
Como se ve en la Fig.3, tenemos 2 capas de entrada (los datos de “Poder” y “Angulo”), 1 capa oculta con 5 salidas y 2 capas de salida (salida final).

Cálculo de las salidas en cada capa:





Función sigmoide entre 0 y 1

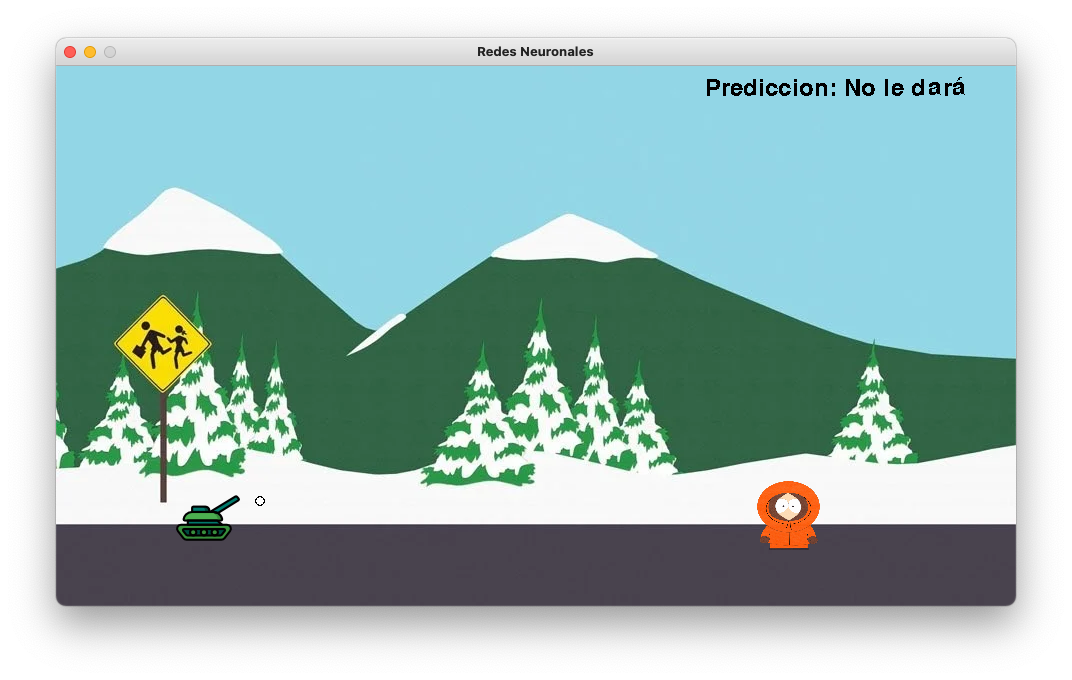


Salida final de la red neuronal

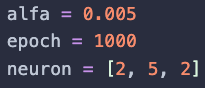
1. **Detalles del código**

La única librería que se usó en el trabajo es Pygame (librería estándar de Python diseñada para juegos) por la posibilidad que nos brindó de dibujar la “puntería” y “bala” del tanque para calcular y realizar el tiro.

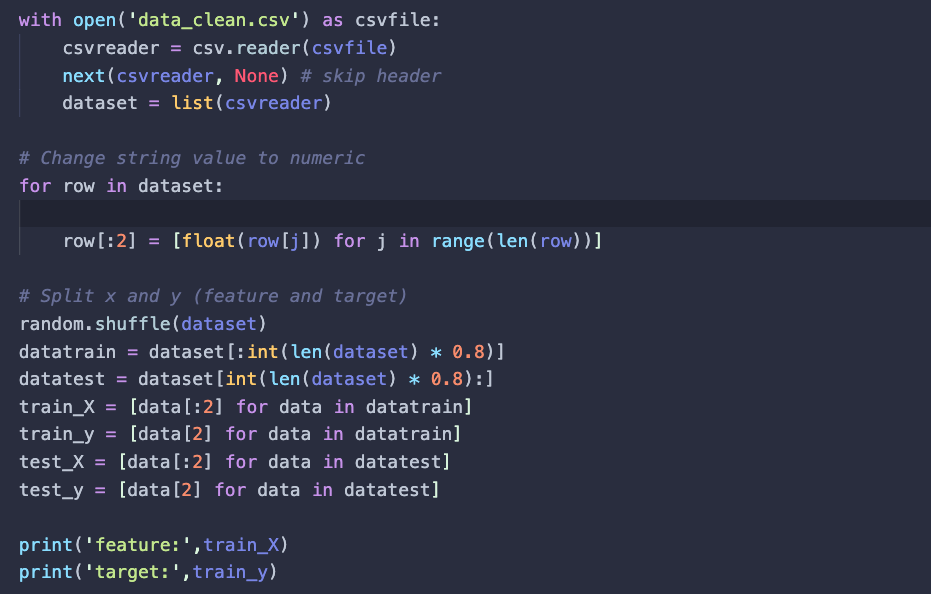
1. **Pruebas**



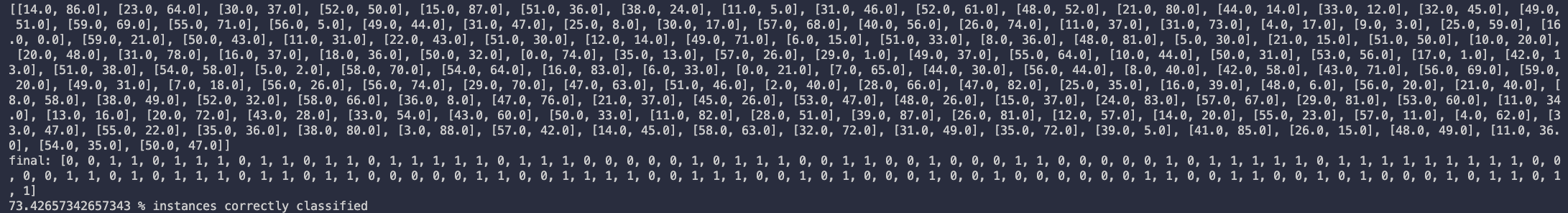
Ejemplo de prueba, el proyectil es muy débil para llegar al objetivo



Parametros



Dataset



Usando el dataset de testeo tenemos 73% de predecir correctamente si el proyectil llegará al objetivo

1. **Referencias**

<https://github.com/renzodamgo/Parabolico> (Repositorio del código)

Anónimo. (2017). Perceptron multicapa. 7 de noviembre del 2020, de Bibing Sitio web: http://bibing.us.es/proyectos/abreproy/12166/fichero/Volumen+1+-+Memoria+descriptiva+del+proyecto%252F3+-+Perceptron+multicapa.pdf