****

ALUMNOS: GONZALEZ OCHOA BRAYANT IVAN, PACHECO RUIZ ISSAC

DATOS GENERALES:

INSTITUTO TECNOLOGICO DE CULIACAN  
HORARIO: 12-13  
CARRERA: INGENIERIA EN SISTEMAS COMPUTACIONALES  
PROFESOR: ZURIEL DATHAN MORA FELIX

tarea 1 unidad 4

TOPICOS DE INTELIGENCIA ARTIFICIAL

## Estructura del Código Principal

El archivo principal carga modelos previamente entrenados almacenados en archivos h5 y los utiliza para predecir el resultado de operaciones dadas. Las operaciones soportadas son las básicas como lo son suma, resta, multiplicación y división,( +, -, \*, /).

## Funcionamiento

* Carga cuatro modelos de IA entrenados (uno por operación).
* Tiene una función llamada calcular que recibe:
* El tipo de operación (+, -, \*, /)
* Dos números (por ejemplo: 3 y 5)
* Según la operación, envía los números al modelo correspondiente.
* Muestra el resultado en pantalla.

### Carga de Modelos

Se importan las librerías necesarias y se cargan cuatro modelos distintos:  
- `modelo\_suma.h5`  
- `modelo\_resta.h5`  
- `modelo\_multi.h5`  
- `modelo\_div.h5`  
Cada modelo ha sido entrenado para aprender una operación específica.

### Función calcular(op, a, b)

Esta función recibe tres parámetros: la operación a realizar (`op`) y dos números `a` y `b`. Se construye una entrada en forma de arreglo NumPy con `[[a, b]]`, que es la entrada esperada por los modelos.  
  
Dependiendo de la operación que queremos realizar, se llama al modelo para predecir el resultado.

Por ejemplo, para una suma (op = '+') se hace:  
`return float(model\_suma.predict(entrada, verbose=0))`  
  
Para la división se agrega una condición extra para evitar división entre cero:  
`return float(model\_div.predict(entrada, verbose=0)) if b != 0 else 'indefinido'`

### Pruebas del Sistema

Al final del archivo se ejecutan ejemplos de cada operación:  
- `3 + 5` → salida del modelo de suma  
- `7 - 2` → salida del modelo de resta  
- `4 \* 6` → salida del modelo de multiplicación  
- `8 / 2` → salida del modelo de división  
- `5 / 0` → retorna 'indefinido' por división entre cero

## Conclusión

A lo largo del desarrollo de la tarea, pudimos ver que se entrenaron modelos de redes neuronales. No se les dijo directamente cómo sumar o restar, sino que se les dieron muchos pares de números con su resultado correcto, y los modelos aprendieron a encontrar ese patrón. Después estos modelos se cargaron en el archivo principal y se utilizaron para predecir resultados nuevos con datos que no habían visto antes.