

Practica 12

Tema: Red neuronal.

A 31 de octubre del 2017.

Introducción.

Una red neuronal es un es un modelo informatico que permite encontrar en un determinado numero de parámetros la combinación ideal para aplicarla al mismo tiempo ya que se ajustan al problema, dados unos parámetros hay una forma de combinarlos para predecir un cierto resultado.

Cuando se usa una red neuronal, se comienza con una unidad fundamental llamada perceptrón, la cual es un elemento que tiene varias entradas con un cierto valor para cada una, por lo tanto, si la suma de cada entrada (n_1 y n_2) por cada valor es mayor que un número determinado (wn), entonces la salida del perceptrón será uno (nf), más sin embargo si la suma de cada entrada (n_1 y n_2) por cada valor es menor que un número determinado (wn), entonces la salida del perceptrón será cero (nf). Tal como se puede representar en la Figura 1.

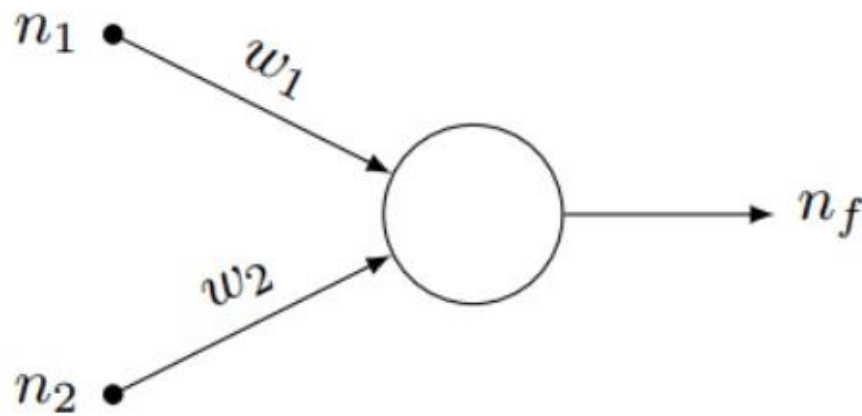


Figura 1: Concepto del perceptrón.

Resumen:

En la práctica No. 12 se pretende simular un modelo de aprendizaje a partir del concepto de las redes neuronales, en el cual el perceptrón tomara una acción a partir de un entrenamiento previo, para posteriormente evaluar y entregar un resultado en el cual puede ser afirmativo o negativo, por lo que estará determinada a la variabilidad de la toma de decisiones que considere pertinentes.

Objetivos:

1. Paralelizar la simulación original y observar los efectos, así como las diferencia en los tiempos de ejecución.
2. Estudiar de manera sistemática el desempeño de la red neuronal para 10 dígitos en función de 3 posibilidades asignadas.

Procedimiento.

Para el primer objetivo, se solicita la paralelización del código debido a que es necesario optimizar los tiempos de ejecución del código para mejorar el desempeño de la simulación, por los tanto los resultados se mostraran en la Figura 2.

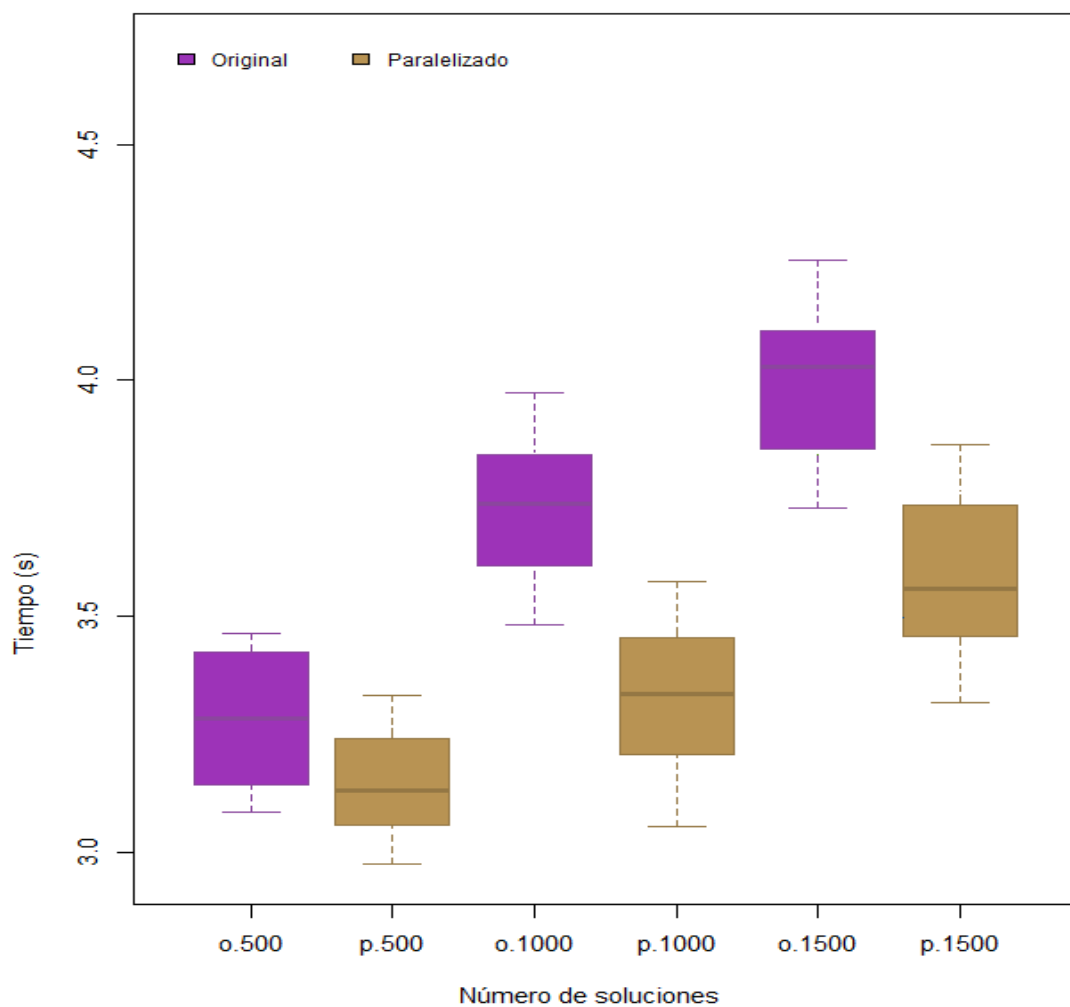


Figura 2. Comparación de tiempos de ejecución

Para esta paralelización se generaron 2 archivos diferentes, un contenía el código original y el otro modificado en la sintaxis paralelizado, esto con el fin de comparar los tiempos de ejecución mediante el siguiente código. Figura 3.

```
for (a in seg(500,1000,1500)) {  
  for (r in 1:20){  
    source('~/.P12/P12/codigo T12/t12p.R', encoding = 'UTF-8')  
    Tpara-cbind(a,r,"o",Tiempo,Acierto)  
  
    source('~/.P12/P12/codigo T12/t12orig.R', encoding = 'UTF-8')  
    Torig-cbind(a,r,"o",Tiempo,Acierto)  
    total-rbind(total,Tpara,Torig)  
  }  
}
```

Figura 3. Código para comparar las simulación original con el paralelo

Conclusiones.

Una vez realizada la comparación, comprendemos que los códigos realizados por el método de paralelización, nos permite optimizar los tiempos de entrega de una respuesta, ya que, al dividir los segmentos simulados respecto a los recursos del equipo, el tiempo de entrega de resultado es mejor.